Optimal constrained interpolation in mesh-adaptive finite element modelling

SIAM J. SCI. COMPUT.
Vol. 39, No. 5, pp. A2257–A2286

OPTIMAL CONSTRAINED INTERPOLATION IN MESH-ADAPTIVE FINITE ELEMENT MODELING*

J. R. MADDISON† AND H. R. HIESTER‡

Abstract. Mesh-to-mesh Galerkin L^2 projection allows piecewise polynomial unstructured finite element data to be interpolated between two nonmatching unstructured meshes of the same domain. The interpolation is by definition optimal in an L^2 sense, and subject to fairly weak assumptions conserves the integral of an interpolated function. However other properties, such as the L^2 norm, or the weak divergence of a vector-valued function, can still be adversely affected by the interpolation. This is an important issue for calculations in which numerical dissipation should be minimized, or for simulations of incompressible flow. This paper considers extensions to mesh-to-mesh Galerkin L^2 projection which are L^2 optimal and ensure exact conservation of key discrete properties, including preservation of both the L^2 norm and the integral, and preservation of both the L^2 norm and weak incompressibility. The accuracy of the interpolants is studied. The utility of the interpolants is studied via adaptive mesh simulations of the two-dimensional lock-exchange problem, which are simulated using a combination of Fluidity and the FEniCS system.

Key words. finite element method, mesh adaptivity, interpolation, lock-exchange

AMS subject classifications. 65M60, 65M50, 46B70

DOI. 10.1137/15M102054X

1. Introduction.

A key characteristic of the finite element method is its ability to utilize complex unstructured meshes. An unstructured mesh finite element numerical model can conform naturally to complex bounding topography without piecewise constant "staircase" boundaries [1], and can locally vary the mesh resolution so as to target regions of relatively increased dynamical importance. On the other hand, unstructured finite element numerical models are typically more expensive than structured alternatives (see, e.g., [29, 76]). In research fields in which structured methods are well established, such as numerical ocean modeling [49, 48], in order for unstructured models to compete it is required that either new unstructured grid optimizations be developed (see, e.g., [64] for optimizations which can be applied for grids with structure in one dimension), or that the approach enable entirely new problems to be tackled, or that accurate boundary representation or a selective use of resolution be able to overcome the relatively increased computational cost per degree of freedom.

A particularly powerful approach, which both utilizes the generality of unstructured meshes and has the potential to overcome the associated increased costs, is to apply dynamic mesh adaptivity (see, e.g., [70, 13, 72, 32, 14]). Here the model mesh is altered in response to the evolving solution, with the aim being to reduce the numerical error per degree of freedom and thereby decrease the computational cost per unit error.

Mesh adaptivity can be applied via several approaches, including modification of

*Submitted to the journal's Methods and Algorithms for Scientific Computing section May 7, 2015; accepted for publication (in revised form) December 2, 2016; published electronically September 28, 2017.

http://www.siam.org/journals/sisc/39-5/M102054.html

Funding: This work was supported by the Engineering and Physical Sciences Research Council under grant EP/G036136/1.

†School of Mathematics and Maxwell Institute for Mathematical Sciences, The University of Edinburgh, Edinburgh EH9 3FD, United Kingdom (j.r.maddison@ed.ac.uk).

‡Center for Oceanic-Atmospheric Prediction Studies, Florida State University, Tallahassee, FL 32306-2741 (hhiester@coaps.fsu.edu).

the mesh topology ("h-adaptivity"), movement of the mesh vertices ("r-adaptivity"), or a redefinition of the solution space by changing the selection of local element basis ("p-adaptivity"). In this paper a form of mixed hr-adaptivity is considered (see, e.g., [71, 75]). In this approach the model equations are integrated on a given mesh for a specified period of time. After this integration time a new optimized mesh is generated based upon estimators or indicators of the local error, and these are used to construct a new mesh by modifying the old through local topological operations and vertex movement. Finally, functions are interpolated from the old "donor" mesh onto the new, optimized "target" mesh, after which the cycle repeats.

The key step considered in this paper is the interpolation of the solution from the donor mesh onto the target mesh. The dynamical component of the numerical model, which performs the numerical integration, is typically carefully designed so as to preserve essential properties, such as exact conservation principles, L^2 stability, or the ability to represent important physical balances. Interpolation of data from the donor mesh to the target can easily destroy these properties, resulting in, for example, a loss of conservation [37] or a loss of dynamical balance [65]. More seriously, interpolation of data may increase function norms and lead to a loss of numerical stability.

In [37] the Galerkin L^2 projection of functions between unstructured meshes is described. The approach is a higher order generalization of the method described in [45] and relies upon the ability to construct a "supermesh," which is an intersection of "donor" and "target" meshes. Specifically, a supermesh of donor and target meshes is constructed so that it is able to represent functions which are polynomials within the elements of either one of the meshes [35], thereby enabling the evaluation of "multimesh" inner products between such donor and target functions. This allows the L^2 projection of a donor function onto a function space defined using the target mesh. Provided the unity element is a member of the target function space, this projection additionally preserves the integral, thereby yielding an L^2 optimal globally conservative projection.

In practice a supermesh is typically very large, and hence it may be infeasible to construct a complete supermesh in memory all at once. This is addressed in [36] via a "local supermeshing" approach, where the intersections of pairs of donor and target elements are constructed individually, integrated over as required, and then discarded (see also [40] and [41]). This simultaneously reduces the memory burden associated with supermeshing, and also reduces the problem of intersecting meshes to the problems of identifying intersecting donor and target elements, and intersecting these elements.

While originally designed for use in dynamic-mesh-adaptive modeling, local supermeshing has since been applied for the calculation of unstructured mesh diagnostics [35, 66], for embedding moving meshes via the "arbitrary mesh interface" method [87], for coupling in fluid-structure interaction problems [86], and in immersed body modeling [19].

In this paper, Galerkin L^2 projection is extended to form *constrained* interpolants. In particular, L^2 optimal interpolants which preserve the L^2 norm and weak incompressibility are considered. Crucially, once mesh-to-mesh Galerkin L^2 projection is available, it is possible to form these constrained interpolants without any further need for supermeshing—the constrained interpolants take the form of simple post-processing steps which can be applied following interpolation via Galerkin L^2 projection.

Constrained projections are already widely used with function spaces defined using a *single* mesh. For example, incompressible Navier–Stokes solvers using a pro-

jection method often perform an L^2 weakly solenoidal projection in order to enforce weak incompressibility (see, e.g., [50]). In [15] a solenoidal interpolant with approximate L^2 norm preservation is constructed, via interpolation of the Helmholtz decomposition of the donor function. In [20] an L^2 optimal weakly solenoidal mesh-to-mesh projection is defined. Such a projection is applied with a low order $P0$–$P1_{CV}$ discretization for the velocity and pressure in [89]. In this paper, the weakly solenoidal projection is considered and analyzed for discretizations with particular mimetic properties, such as the $P1_{DG}$–$P2$ finite element pair, and is then extended to form an L^2 norm preserving weakly solenoidal interpolant.

The paper proceeds as follows. Section 2 describes mesh-to-mesh Galerkin L^2 projection and its error and dissipation properties. L^2 norm preserving interpolants are described in section 3. Section 4 considers mesh-to-mesh weakly solenoidal projection for discretizations with specific mimetic properties ("gradient embedding" and "skew-gradient embedding"), and this is combined with the norm preserving interpolant to yield an L^2 optimal L^2 norm preserving weakly solenoidal interpolant. The interpolants are tested in section 5 via dynamic-mesh-adaptive simulations of the two-dimensional lock-exchange problem. The paper concludes in section 6.

2. Galerkin L^2 projection.

In this section, Galerkin L^2 projection (hereafter "Galerkin projection") of unstructured mesh data via supermesh construction is described. The principles of mesh-to-mesh Galerkin projection are outlined in section 2.1, and the relation between the L^2 error and dissipation of the L^2 norm is described in section 2.2.

2.1. Mesh-to-mesh Galerkin projection.

Mesh-to-mesh Galerkin projection via supermesh construction is described in [37, 36] and in further detail in [34]. Since this paper describes an extension to this method, the key principles are outlined here.

Throughout this paper, real functions which are polynomials within the elements of a simplex mesh are considered. Specifically, let $\Omega \subset \mathbb{R}^d$ refer to a d-dimensional open bounded domain with Lipschitz boundary $\partial\Omega$. Simplex meshes are considered. Here a simplex mesh $\mathcal{T}(\Omega)$ is a triangulation of the domain $\Omega \cup \partial\Omega$ satisfying properties $(\mathcal{T}_h 1)$–$(\mathcal{T}_h 4)$ of [23]. In particular, each element $e \in \mathcal{T}(\Omega)$ is a simplex with nonzero volume, and the elements are nonoverlapping and cover Ω.

The discrete function space $\mathcal{P}_n(\mathcal{T}) \subset L^2(\Omega)$ is defined to be the space of functions each of which are, within each element of $\mathcal{T}(\Omega)$, a polynomial with maximal degree n. In standard finite element notation $\mathcal{P}_n = Pn_{DG}$, $\mathcal{P}_0 = P0$, and for conforming meshes (meshes additionally satisfying property $(\mathcal{T}_h 5)$ of [23]) $\mathcal{P}_n \cap H^1(\Omega) = Pn$. Crucially, there exist accurate and efficient methods for integrating piecewise polynomial functions over simplex elements (see, e.g., [81] and [23, section 4.1]). Hence, for such meshes, the L^2 inner product of two elements in $\mathcal{P}_n$ can be computed efficiently via numerical integration.

Now consider Galerkin projection between two real Hilbert spaces.

DEFINITION 2.1 (Galerkin projection). *Given real Hilbert spaces $V_D, V_T \subseteq L^2(\Omega)$, the Galerkin projection is the map $\Pi : V_D \to V_T$ such that for each $\phi_D \in V_D$,*

$$\Pi(\phi_D) = \arg \min_{\Phi_T \in V_T} \|\Phi_T - \phi_D\|_{L^2}. \tag{1}$$

It follows that the map Π satisfies (see, e.g., [36] and [46, Appendix A.3.2.1])

$$\langle \zeta_T, \Pi(\phi_D) \rangle_{L^2} = \langle \zeta_T, \phi_D \rangle_{L^2} \quad \forall \zeta_T \in V_T. \tag{2}$$

If the unity element $1 \in V_T$, then Galerkin projection is globally conservative, in the sense that

$$\langle 1, \Pi(\phi_D) \rangle_{L^2} = \langle 1, \phi_D \rangle_{L^2}. \tag{3}$$

A critical technical issue appears when considering mesh-to-mesh Galerkin projection. Consider a "donor" simplex mesh $\mathcal{T}_D(\Omega)$, a "target" simplex mesh $\mathcal{T}_T(\Omega)$, and discrete function spaces $V_D \subseteq \mathcal{P}_m(\mathcal{T}_D)$ and $V_T \subseteq \mathcal{P}_n(\mathcal{T}_T)$. In this case the right-hand side of (2) involves a "multimesh" inner product involving elements defined on the donor and target meshes. The issue is clarified by equipping V_T with a finite basis, $\{\zeta_{T,1}, \zeta_{T,2}, \ldots, \zeta_{T,N_T}\}$, which from (2) leads to the linear system

$$M\tilde{\phi}_T = b, \tag{4}$$

where

$$M_{ij} = \langle \zeta_{T,i}, \zeta_{T,j} \rangle_{L^2}, \tag{5a}$$

$$\Pi(\phi_D) = \sum_{i=1}^{N_T} \tilde{\phi}_{T,i} \zeta_{T,i}, \tag{5b}$$

$$b_i = \langle \zeta_{T,i}, \phi_D \rangle_{L^2}. \tag{5c}$$

The mass matrix M can be assembled on the target mesh. The issue is that assembly of the right-hand side vector b requires evaluation of multimesh inner products. In general neither mesh is suitable for performing the numerical integration required in order to assemble b [36].

This issue is addressed via the construction of a supermesh.

DEFINITION 2.2. *Given two meshes $\mathcal{T}_D(\Omega)$ and $\mathcal{T}_T(\Omega)$, a supermesh $\mathcal{T}_S(\Omega)$ is a mesh where each element of the supermesh is contained within one and only one element of the donor mesh $\mathcal{T}_D(\Omega)$, and is also contained within one and only one element of the target mesh $\mathcal{T}_T(\Omega)$. That is,*

$$\forall e_S \in \mathcal{T}_S(\Omega)\, \exists! e_D \in \mathcal{T}_D(\Omega) \;\; s.t.\; e_S \subseteq e_D, \tag{6a}$$

$$\forall e_S \in \mathcal{T}_S(\Omega)\, \exists! e_T \in \mathcal{T}_T(\Omega) \;\; s.t.\; e_S \subseteq e_T. \tag{6b}$$

This definition is equivalent to Lemma 2 of [37]. Note that a supermesh of two meshes is not uniquely defined. Crucially, a supermesh can be used to construct a superspace of donor and target spaces.

LEMMA 2.3. *Consider a simplex supermesh $\mathcal{T}_S(\Omega)$ of two simplex meshes $\mathcal{T}_D(\Omega)$ and $\mathcal{T}_T(\Omega)$. Then*

$$\left\{\zeta_D \zeta_T \in L^2(\Omega) : \zeta_D \in \mathcal{P}_m(\mathcal{T}_D), \zeta_T \in \mathcal{P}_n(\mathcal{T}_T)\right\} \subseteq \mathcal{P}_{m+n}(\mathcal{T}_S). \tag{7}$$

Proof. This follows immediately from Lemma 1 of [35]. ∎

Hence, given $\zeta_D \in \mathcal{P}_m(\mathcal{T}_D)$ and $\zeta_T \in \mathcal{P}_n(\mathcal{T}_T)$, $\zeta_D \zeta_T \in \mathcal{P}_{m+n}(\mathcal{T}_S)$. The right-hand side vector b can therefore be assembled via numerical integration over a supermesh.

A triangle supermesh is shown in Figure 1. In practice, construction of a supermesh requires the solution of a computational geometry problem: the identification of pairs of elements $e_D \in \mathcal{T}_D$ and $e_T \in \mathcal{T}_T$ which intersect, and the construction of a

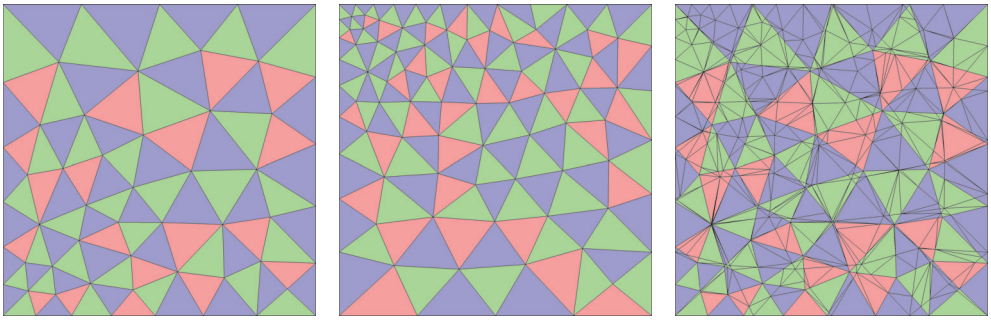


FIG. 1. *Left and middle: Two triangle meshes for the unit square generated using Gmsh* [43]*, with* 86 *elements and* 146 *elements, respectively. Right: A triangle supermesh generated using the supermeshing functionality of the Fluidity code* [73, 30] *using the approach of* [36]*, colored according to the elements of the left mesh and with* 778 *nondegenerate elements.*[1] *See also Figure* 1 *of* [37].

mesh of their intersection. Moreover, the complete supermesh is typically very large, necessitating an interleaving of the processes of generating the supermesh and integrating over it. The resulting algorithm is referred to as "local supermeshing" and is described in [36].

2.2. Error and dissipation. First, the error in a general mesh-to-mesh interpolant is considered.

LEMMA 2.4. *Consider real Hilbert spaces* $V_D, V_T \subseteq L^2(\Omega)$ *and a map* $\mathcal{I} : V_D \to V_T$*. Then*

$$\|\mathcal{I}(\phi_D) - \phi_D\|_{L^2}^2 = \|\mathcal{I}(\phi_D)\|_{L^2}^2 + \|\phi_D\|_{L^2}^2 - 2\langle \mathcal{I}(\phi_D), \Pi(\phi_D)\rangle_{L^2}, \tag{8}$$

where $\Pi(\phi_D)$ *is the Galerkin projection of* ϕ_D *onto* V_T*.*

Proof. This follows directly from the expansion of the L^2 norm and use of (2). ∎

This basic property has an important consequence: when $V_D \subseteq \mathcal{P}_m(\mathcal{T}_D)$ and $V_T \subseteq \mathcal{P}_n(\mathcal{T}_T)$ are function spaces defined on the donor and target meshes, respectively, then it follows that if the Galerkin projection $\Pi(\phi_D)$ is available, the second right-hand side term in (8) can be evaluated via numerical integration over the donor mesh, and the remaining two terms can be evaluated via numerical integration over the target mesh. That is, the error norm $\|\mathcal{I}(\phi_D) - \phi_D\|_{L^2}$ can be computed without any further multimesh integration.

In [36] interpolation error norms are evaluated by explicitly forming the error $\mathcal{I}(\phi_D) - \phi_D$ on the supermesh, and by performing the necessary numerical integration over the supermesh. However, if the L^2 error is to be computed and the Galerkin projection is available, then this additional multimesh integration step is not required.

Dissipation of the L^2 norm by Galerkin projection can be bounded in terms of the L^2 projection error.

LEMMA 2.5. *Consider real Hilbert spaces* $V_D, V_T \subseteq L^2(\Omega)$ *and the Galerkin projection* $\Pi : V_D \to V_T$*. Let* $\varepsilon_\Pi(\phi_D) = \|\Pi(\phi_D) - \phi_D\|_{L^2}$ *be the* L^2 *error in the Galerkin*

[1] The local supermeshing method of [36] implemented in Fluidity in this example generates 6 additional numerically degenerate elements, with a minimum edge length less than 6×10^{-17}. Note that via (5) the mass matrix can be assembled via integration only over the target mesh. The near-degenerate elements lead to small contributions to the right-hand side of the linear system but do not lead to any particular difficulties regarding the linear system matrix.

projection. Then

$$\frac{1}{2}\varepsilon_\Pi(\phi_D)^2 \leq \left[\|\phi_D\|_{L^2} - \|\Pi(\phi_D)\|_{L^2}\right]\|\phi_D\|_{L^2} \leq \varepsilon_\Pi(\phi_D)^2. \tag{9}$$

Proof. Pythagoras' theorem (which follows from Lemma 2.4 with $\mathcal{I} = \Pi$; see also [46, Appendix A.3.2.1]) yields

$$\|\phi_D\|_{L^2}^2 - \|\Pi(\phi_D)\|_{L^2}^2 = \varepsilon_\Pi(\phi_D)^2, \tag{10}$$

and factorizing the left-hand side leads to

$$\left[\|\phi_D\|_{L^2} - \|\Pi(\phi_D)\|_{L^2}\right]\left[\|\phi_D\|_{L^2} + \|\Pi(\phi_D)\|_{L^2}\right] = \varepsilon_\Pi(\phi_D)^2. \tag{11}$$

The bounds then follow from $\|\phi_D\|_{L^2} \leq \|\phi_D\|_{L^2} + \|\Pi(\phi_D)\|_{L^2}$ and, using (10), $\|\phi_D\|_{L^2} + \|\Pi(\phi_D)\|_{L^2} = \|\phi_D\|_{L^2} + \sqrt{\|\phi_D\|_{L^2}^2 - \varepsilon_\Pi(\phi_D)^2} \leq 2\|\phi_D\|_{L^2}$. ∎

Hence Galerkin projection is dissipative, in the sense that it decreases the L^2 norm of an interpolated function whenever the L^2 interpolation error is nonzero. Moreover, the dissipation is small, being bounded in terms of the square of this error. This is a potentially extremely useful property—if a numerical model is constructed so as to be L^2 stable, then a mesh-adaptive version of the model utilizing mesh-to-mesh Galerkin projection will remain L^2 stable. This need not be the case for alternative interpolants, where stability of the model can potentially be compromised by the mesh-to-mesh interpolation.

3. L^2 norm preserving interpolants. While Galerkin projection decreases the L^2 norm by only a small amount, as expressed by (9), there may be cases in which even this is unacceptable. For example, in very long ocean or atmosphere calculations, spurious injection and dissipation of energy or enstrophy may affect the resulting solutions, or their turbulent spectra [77]. Numerical schemes are often carefully designed so as to observe key conservation laws (see, e.g., [5, 38, 78]), and without additional steps these conservation principles are easily destroyed by mesh-to-mesh interpolation in a mesh-adaptive numerical model.

This issue is addressed here via the construction of interpolants which preserve the L^2 norm. The L^2 optimal L^2 norm preserving interpolant is derived in section 3.1. This interpolant is not in general globally conservative even if the unity element is in the target function space. An alternative interpolant is therefore derived in section 3.2, yielding the L^2 optimal L^2 norm preserving conservative interpolant. In each case, the constrained interpolants are simple modifications to Galerkin projection and have an L^2 error which can be related to the Galerkin projection L^2 error.

Note that much of the discussion of this section generalizes to allow the formulation of interpolants which are optimal in alternative norms. Here extensions to mesh-to-mesh Galerkin projection are sought, and hence L^2 optimal interpolants are considered.

3.1. L^2 norm preserving interpolation. The general principal of the L^2 norm preserving interpolant is illustrated in Figure 2. The Galerkin projection is an orthogonal projection resulting, for a nonzero projection error, in a projected function with a smaller norm than the original donor function. In finite dimensions the functions with norm equal to the donor function norm lie on a sphere. The L^2 optimal L^2 norm preserving interpolant is the point on this sphere which is in the same direction as the projected function, but with an appropriately increased magnitude.

This can be derived via the introduction of a nonlinear constraint.

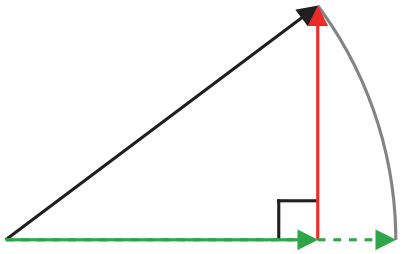


FIG. 2. *Schematic of the L^2 norm preserving interpolant $\mathcal{I}_1$. An orthogonal projection of the donor data (diagonal black vector) onto a horizontal line yields the optimal orthogonal projection (horizontal green vector) with an orthogonal error (vertical red vector). Norm preservation is restored by extending this projection (by the dashed green vector).*

LEMMA 3.1. *Consider real Hilbert spaces $V_D, V_T \subseteq L^2(\Omega)$ and the Galerkin projection $\Pi : V_D \to V_T$. For each $\phi_D \in V_{D,1} = \{\zeta_D \in V_D : \|\Pi(\zeta_D)\|_{L^2} \neq 0\}$ there exists a unique $\phi_T \in V_{T,1,\phi_D} = \{\zeta_T \in V_T : \|\zeta_T\|_{L^2} = \|\phi_D\|_{L^2}\}$ such that*

$$\|\phi_T - \phi_D\|_{L^2} \leq \|\Phi_T - \phi_D\|_{L^2} \quad \forall \Phi_T \in V_{T,1,\phi_D}. \tag{12}$$

This defines a map $\mathcal{I}_1 : V_{D,1} \to V_T$ where

$$\mathcal{I}_1(\phi_D) = \arg \min_{\Phi_T \in V_{T,1,\phi_D}} \|\Phi_T - \phi_D\|_{L^2}. \tag{13}$$

Proof. Given a $\phi_D \in V_{D,1}$ define the functional $J_1 : V_{T,1,\phi_D} \to \mathbb{R}$ where

$$J_1(\Phi_T) = \frac{1}{2}\|\Phi_T - \phi_D\|_{L^2}^2. \tag{14}$$

Define the Lagrange constrained functional $\hat{J}_1 : V_T \times \mathbb{R} \to \mathbb{R}$ where

$$\hat{J}_1(\Phi_T, \Lambda) = \frac{1}{2}\|\Phi_T - \phi_D\|_{L^2}^2 + \frac{\Lambda}{2}\left[\|\Phi_T\|_{L^2}^2 - \|\phi_D\|_{L^2}^2\right], \tag{15}$$

and seek critical points $(\phi_T, \lambda) \in V_T \times \mathbb{R}$ at which the Gâteaux derivative vanishes as follows:

$$\mathrm{d}\hat{J}_1(\phi_T, \lambda; \zeta_T, 0) = \langle \zeta_T, (1+\lambda)\phi_T - \phi_D \rangle_{L^2} = 0 \quad \forall \zeta_T \in V_T, \tag{16a}$$

$$\mathrm{d}\hat{J}_1(\phi_T, \lambda; 0, 1) = \frac{1}{2}\|\phi_T\|_{L^2}^2 - \frac{1}{2}\|\phi_D\|_{L^2}^2 = 0. \tag{16b}$$

By inspection, $(1+\lambda)\phi_T = \Pi(\phi_D)$. It then follows (for example, via direct substitution or, more formally, via application of the Lagrange multiplier theorem; see, e.g., [17, section 11.3]) that minimization of J_1 defines a map $\mathcal{I}_1 : V_{D,1} \to V_T$ where

$$\mathcal{I}_1(\phi_D) = \gamma \Pi(\phi_D), \tag{17}$$

with scaling factor

$$\gamma = \frac{\|\phi_D\|_{L^2}}{\|\Pi(\phi_D)\|_{L^2}}. \tag{18}$$

∎

If $\phi_D \in V_D$ but $\phi_D \notin V_{D,1}$, then the donor function ϕ_D is L^2 orthogonal to the target function space V_T, and the analogous optimization problem does not in general have a unique solution.

If the mesh-to-mesh Galerkin projection is available, then the scaling factor γ can be computed via numerical integration over the donor and target meshes, without any further multimesh integration. Hence the L^2 norm preserving interpolation can be performed via simple postprocessing following mesh-to-mesh Galerkin projection. It is also worth noting that due to Lemma 2.5, the factor γ is greater than or equal to one, which could be a concern if preservation of the data bounds is required.

The error in the constrained interpolant can be directly related to the Galerkin projection error.

LEMMA 3.2. *The L^2 norm preserving interpolant $\mathcal{I}_1$ has an L^2 error $\varepsilon_{\mathcal{I}_1}(\phi_D)$ which can be bounded as follows in terms of the Galerkin projection L^2 error, $\varepsilon_\Pi(\phi_D)$:*

$$\varepsilon_{\mathcal{I}_1}(\phi_D) \leq \sqrt{2}\varepsilon_\Pi(\phi_D)\,, \tag{19a}$$

and which has the following asymptotic expansion:

$$\varepsilon_{\mathcal{I}_1}(\phi_D) = \varepsilon_\Pi(\phi_D) + \frac{1}{8}\frac{\varepsilon_\Pi(\phi_D)^3}{\|\phi_D\|_{L^2}^2} + \mathcal{O}\left(\frac{\varepsilon_\Pi(\phi_D)^5}{\|\phi_D\|_{L^2}^4}\right) \text{ as } \frac{\varepsilon_\Pi(\phi_D)^5}{\|\phi_D\|_{L^2}^4} \to 0. \tag{19b}$$

Proof. $\|\mathcal{I}_1(\phi_D)\|_{L^2} = \|\phi_D\|_{L^2}$, (8), (10), and (17) lead to

$$\begin{aligned}
\varepsilon_{\mathcal{I}_1}(\phi_D)^2 &= \|\mathcal{I}_1(\phi_D)\|_{L^2}^2 + \|\phi_D\|_{L^2}^2 - 2\langle \mathcal{I}_1(\phi_D), \Pi(\phi_D)\rangle_{L^2} \\
&= 2\|\phi_D\|_{L^2}^2 - 2\|\phi_D\|_{L^2}\|\Pi(\phi_D)\|_{L^2} \\
&= 2\|\Pi(\phi_D)\|_{L^2}^2 + 2\varepsilon_\Pi(\phi_D)^2 - 2\|\phi_D\|_{L^2}\|\Pi(\phi_D)\|_{L^2} \\
&\leq 2\varepsilon_\Pi(\phi_D)^2\,, \qquad (20)
\end{aligned}$$

which yields (19a). It further follows that

$$\begin{aligned}
\varepsilon_{\mathcal{I}_1}(\phi_D)^2 &= 2\|\phi_D\|_{L^2}^2 - 2\|\phi_D\|_{L^2}\|\Pi(\phi_D)\|_{L^2} \\
&= 2\|\phi_D\|_{L^2}^2\left(1 - \sqrt{1 - \frac{\varepsilon_\Pi(\phi_D)^2}{\|\phi_D\|_{L^2}^2}}\right), \qquad (21)
\end{aligned}$$

and Taylor expansion then yields (19b). ∎

That is, the L^2 norm preserving interpolant is a higher order modification to Galerkin projection, with an L^2 error which is in all cases bounded by $\sqrt{2}$ times the Galerkin projection L^2 error, and which has the same leading asymptotic L^2 error.

3.2. L^2 norm preserving conservative interpolation. While Galerkin projection Π is globally conservative if the unity element is in the target space $1 \in V_T$, the L^2 norm preserving interpolant $\mathcal{I}_1$ generally is not: the addition of L^2 norm preservation is achieved at the price of a loss of global conservation (see, for example, Figure 3 and Table 1). The L^2 norm preserving interpolant (17) is globally conservative only if either $\langle 1, \phi_D\rangle_{L^2} = 0$ or the Galerkin projection has zero L^2 error.

Preservation of both the L^2 norm and the global integral is equivalent to preservation of both the L^2 norm and the L^2 inner product with a single element in the target space—specifically the unity element $1 \in V_T$. Initially a more general interpolant is

considered, which preserves both the L^2 norm and the L^2 inner product with a finite number of linearly independent functions in $V_D \cap V_T$. This will later be used in section 5 to define an interpolant which, for a Boussinesq Navier–Stokes calculation with a linear equation of state, preserves all of the temperature L^2 norm, global integral, and potential energy.

LEMMA 3.3. *Consider real Hilbert spaces $V_D, V_T \subseteq L^2(\Omega)$ and the Galerkin projection $\Pi : V_D \to V_T$. Consider a subspace $X \subseteq V_D \cap V_T$ with finite basis, and define the L^2 projection of a donor or target function onto X, $\Pi_X : V_D \cup V_T \to X$ where*

$$\langle \xi, \Pi_X(\zeta) \rangle_{L^2} = \langle \xi, \zeta \rangle_{L^2} \quad \forall \xi \in X. \tag{22}$$

For each $\phi_D \in V_{D,X} = \{\zeta_D \in V_D : \|\Pi(\zeta_D) - \Pi_X(\zeta_D)\|_{L^2} \neq 0\}$ there exists a unique $\phi_T \in V_{T,X,\phi_D} = \{\zeta_T \in V_T : \|\zeta_T\|_{L^2} = \|\phi_D\|_{L^2}, \Pi_X(\zeta_T) = \Pi_X(\phi_D)\}$ such that

$$\|\phi_T - \phi_D\|_{L^2} \leq \|\Phi_T - \phi_D\|_{L^2} \quad \forall \Phi_T \in V_{T,X,\phi_D}. \tag{23}$$

This defines a map $\mathcal{I}_X : V_{D,X} \to V_T$ where

$$\mathcal{I}_X(\phi_D) = \arg \min_{\Phi_T \in V_{T,X,\phi_D}} \|\Phi_T - \phi_D\|_{L^2}. \tag{24}$$

Proof. Given a $\phi_D \in V_{D,X}$ define the functional $J_X : V_{T,X,\phi_D} \to \mathbb{R}$ where

$$J_X(\Phi_T) = \frac{1}{2} \|\Phi_T - \phi_D\|_{L^2}^2. \tag{25}$$

Define the Lagrange constrained functional $\hat{J}_X : V_T \times \mathbb{R} \times X \to \mathbb{R}$ where

$$\hat{J}_X(\Phi_T, \Lambda, M) = \frac{1}{2} \|\Phi_T - \phi_D\|_{L^2}^2 + \frac{\Lambda}{2} \left[\|\Phi_T\|_{L^2}^2 - \|\phi_D\|_{L^2}^2 \right] + \langle M, \Phi_T - \phi_D \rangle_{L^2}, \tag{26}$$

and seek critical points $(\phi_T, \lambda, \mu) \in V_T \times \mathbb{R} \times X$ at which the Gâteaux derivative vanishes as follows:

$$\mathrm{d}\hat{J}_X(\phi_T, \lambda, \mu; \zeta_T, 0, 0) = \langle \zeta_T, (1+\lambda)\phi_T + \mu - \phi_D \rangle_{L^2} = 0 \quad \forall \zeta_T \in V_T, \tag{27a}$$

$$\mathrm{d}\hat{J}_X(\phi_T, \lambda, \mu; 0, 1, 0) = \frac{1}{2} \|\phi_T\|_{L^2}^2 - \frac{1}{2} \|\phi_D\|_{L^2}^2 = 0, \tag{27b}$$

$$\mathrm{d}\hat{J}_X(\phi_T, \lambda, \mu; 0, 0, \xi) = \langle \xi, \phi_T - \phi_D \rangle_{L^2} = 0 \quad \forall \xi \in X. \tag{27c}$$

By inspection, $(1+\lambda)\phi_T + \mu = \Pi(\phi_D)$. It then follows (for example, via direct substitution or, more formally, via application of the Lagrange multiplier theorem; see, e.g., [17, section 11.3]) that minimization of J_X defines a map $\mathcal{I}_X : V_{D,X} \to V_T$ where

$$\mathcal{I}_X(\phi_D) = \frac{\|\phi_D - \Pi_X(\phi_D)\|_{L^2}}{\|\Pi(\phi_D) - \Pi_X(\Pi(\phi_D))\|_{L^2}} [\Pi(\phi_D) - \Pi_X(\Pi(\phi_D))] + \Pi_X(\phi_D). \tag{28}$$

∎

X is here defined to be a subset of both V_D and V_T. The key consequence of this is that the projections onto X appearing in (28) can be performed without any multimesh integration. Hence the interpolant $\mathcal{I}_X$ can be constructed via simple postprocessing steps applied following mesh-to-mesh Galerkin projection.

The error in the constrained interpolation can be directly related to the Galerkin projection error.

LEMMA 3.4. *The interpolant $\mathcal{I}_X$ has an L^2 error $\varepsilon_{\mathcal{I}_X}(\phi_D)$ which can be bounded as follows in terms of the Galerkin projection L^2 error $\varepsilon_\Pi(\phi_D)$:*

$$\varepsilon_{\mathcal{I}_X}(\phi_D) \leq \sqrt{2}\varepsilon_\Pi(\phi_D)\,, \tag{29a}$$

and which has the following asymptotic expansion:

$$\varepsilon_{\mathcal{I}_X}(\phi_D) = \varepsilon_\Pi(\phi_D) + \frac{1}{8}\frac{\varepsilon_\Pi(\phi_D)^3}{\|\phi_D - \Pi_X(\phi_D)\|_{L^2}^2} + \mathcal{O}\left(\frac{\varepsilon_\Pi(\phi_D)^5}{\|\phi_D - \Pi_X(\phi_D)\|_{L^2}^4}\right)$$

$$\text{as } \frac{\varepsilon_\Pi(\phi_D)^5}{\|\phi_D - \Pi_X(\phi_D)\|_{L^2}^4} \to 0. \tag{29b}$$

Proof. Noting that $\Pi(\Pi_X(\phi_D)) = \Pi_X(\Pi(\phi_D))$ and using linearity of the Galerkin projection, the L^2 error associated with $\mathcal{I}_X$ and $\mathcal{I}_1$ can be related as follows:

$$\begin{aligned}
\varepsilon_{\mathcal{I}_X}(\phi_D) &= \|\mathcal{I}_X(\phi_D) - \phi_D\|_{L^2} \\
&= \left\| \frac{\|\phi_D - \Pi_X(\phi_D)\|_{L^2}}{\|\Pi(\phi_D - \Pi_X(\phi_D))\|_{L^2}} \Pi(\phi_D - \Pi_X(\phi_D)) - [\phi_D - \Pi_X(\phi_D)] \right\|_{L^2} \\
&= \|\mathcal{I}_1(\phi_D - \Pi_X(\phi_D)) - (\phi_D - \Pi_X(\phi_D))\|_{L^2} \\
&= \varepsilon_{\mathcal{I}_1}(\phi_D - \Pi_X(\phi_D))\,. \qquad (30)
\end{aligned}$$

The result then follows from Lemma 3.2, $\Pi(\Pi_X(\phi_D)) = \Pi_X(\phi_D)$, and linearity of the Galerkin projection. ∎

An L^2 optimal L^2 norm preserving conservative interpolant is now defined by letting X be the space of constant valued functions, $X = \text{span}(\{1\})$, which defines a map $\mathcal{I}_2 : V_{D,2} \to V_T$ where $V_{D,2} = \left\{\phi_D \in V_D : \left\|\Pi(\phi_D) - \overline{\Pi(\phi_D)}\right\|_{L^2} \neq 0\right\}$ and where

$$\mathcal{I}_2(\phi_D) = \frac{\left\|\phi_D - \overline{\phi_D}\right\|_{L^2}}{\left\|\Pi(\phi_D) - \overline{\Pi(\phi_D)}\right\|_{L^2}} \left(\Pi(\phi_D) - \overline{\Pi(\phi_D)}\right) + \overline{\phi_D}, \tag{31}$$

where here an overbar denotes an average over the domain Ω.

By construction, $\mathcal{I}_2$ is the L^2 optimal L^2 norm preserving conservative interpolation of $\phi_D \in V_{D,2}$ onto V_T. If $\phi_D \in V_D$ but $\phi_D \notin V_{D,2}$, then the analogous optimization does not have a solution, or the solution is not unique. The L^2 norm preserving conservative interpolant is a higher order modification to Galerkin projection, with an L^2 error which is in all cases bounded by $\sqrt{2}$ times the Galerkin projection L^2 error, and which has the same leading asymptotic L^2 error. As before, if the mesh-to-mesh Galerkin projection is available, then the L^2 norm preserving conservative interpolation can be performed via simple postprocessing following Galerkin projection and without any further multimesh integration.

The differing properties of Galerkin projection and the L^2 optimal L^2 norm preserving interpolants are illustrated via the simple analytic example in Figure 3 and Table 1, interpolating the quadratic function $\phi_D = x^2$ onto the space of linear functions on the unit interval $[0, 1]$. The L^2 norm preserving conservative interpolant $\mathcal{I}_2$ has a larger L^2 error than the L^2 norm preserving interpolant $\mathcal{I}_1$, which itself has a larger L^2 error than Galerkin projection Π. However, the additional error introduced by the L^2 norm preserving interpolants is relatively small, given that the L^2 error due to Galerkin projection is rather large, that is, $\varepsilon_\Pi(\phi_D) / \|\phi_D\|_{L^2} = 16.7\%$. The L^2 norm preserving interpolants amplify the under-shoot near $x = 0$, illustrating the compromise made between norm preservation and boundedness.

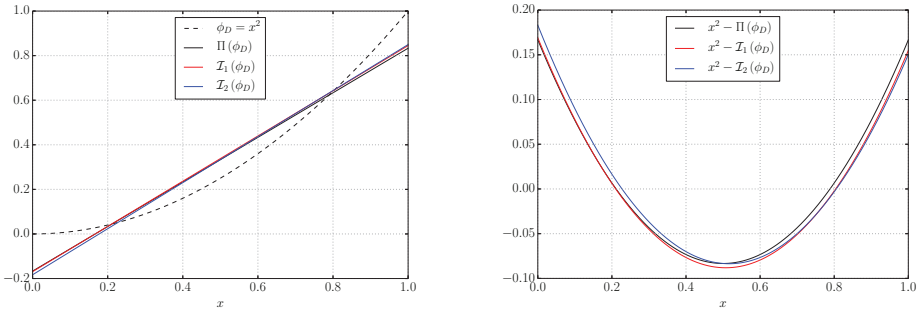


FIG. 3. *Comparison of Galerkin projection and the L^2 norm preserving interpolants $\mathcal{I}_1$ and $\mathcal{I}_2$, interpolating the quadratic function x^2 onto the space of linear functions on the unit interval $[0,1]$. Left: The differing interpolations. Right: Local error.*

TABLE 1

Comparison of Galerkin projection and the L^2 norm preserving interpolants $\mathcal{I}_1$ and $\mathcal{I}_2$, interpolating the quadratic function x^2 onto the space of linear functions on the unit interval $[0,1]$. Decimal approximations are given to five significant figures.

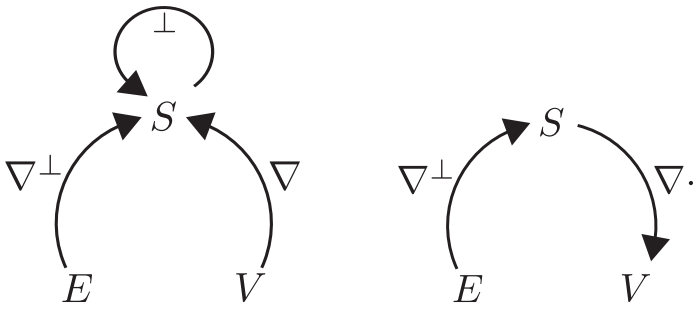
	Integral	L^2 norm	Min. value	Max. value	L^2 error
$\phi_D = x^2$	0.33333	0.44721	0	1	–
$\Pi(\phi_D) = x - \frac{1}{6}$	0.33333	0.44096	−0.16667	0.83333	0.074536
$\mathcal{I}_1(\phi_D) = \frac{6}{\sqrt{35}}\left(x - \frac{1}{6}\right)$	0.33806	0.44721	−0.16903	0.84515	0.074798
$\mathcal{I}_2(\phi_D) = \frac{4}{\sqrt{15}}x + \frac{1}{3} - \frac{2}{\sqrt{15}}$	0.33333	0.44721	−0.18306	0.84973	0.075134

4. Weakly solenoidal interpolation for the $P1_{DG}$–$P2$ finite element pair in two dimensions. Mesh-to-mesh interpolation may lead to a loss of more subtle numerical properties. In particular, dynamic mesh adaptivity has previously been applied for the solution of the time-dependent incompressible Navier–Stokes equations for applications in numerical ocean modeling (see, e.g., [75, 74, 68, 54, 55]). Numerical issues can be expected if the incompressible Navier–Stokes equations are supplied with a *divergent* initial condition for the velocity, as this implies a loss of regularity in time (see [47, sections 3.9 and 3.16] for detailed discussions relating to initializing a Navier–Stokes solver). Similarly, numerical issues might be encountered if, when applying dynamic mesh adaptivity, the interpolated velocities are divergent.

In this section an L^2 optimal interpolant which preserves weak incompressibility is described. Such an interpolant is applied in [89] with a $P0$–$P1_{CV}$ discretization. The interpolant is also as described in [20]—the key differences here are that the target space is not in $H(\Omega, \mathrm{div})$, and additional error bounds are derived for finite element discretizations with specific mimetic properties. The interpolant is further extended to form an L^2 optimal L^2 norm preserving weakly solenoidal interpolant.

4.1. Mimetic properties. In this section finite element discretizations which satisfy key mimetic properties are considered. This significantly simplifies both the analysis and implementation of the mesh-to-mesh interpolants to be described, but note that solenoidal interpolants can be constructed without requiring these properties (see [20]). Further details regarding mimetic finite elements can be found in [6, 28].

Two dimensions and function spaces with specific "embedding properties" are



Cotter and Ham [24]. Cotter and Shipton [27].

FIG. 4. *Left: Mimetic diagram for the finite element pairs described in* [24]*, including the* $P1_{DG}$–$P2$ *finite element pair in two dimensions. The "curl" space* $E \subset H^1(\Omega)$ *skew gradient embeds in* $S \subset \left[L^2(\Omega)\right]^2$*, and the "divergence" space* $V \subset H^1(\Omega)$ *gradient embeds in* S*. S also "perp" embeds into itself (see Definition* 1*, point* 2 *of* [24]*). Right: Mimetic diagram for the finite element pairs described in* [27]*, where* $E \subset H^1(\Omega)$*,* $S \subset H(\Omega, \mathrm{div})$*, and* $V \subset L^2(\Omega)$*. These satisfy skew gradient embedding and a divergence embedding property.*

considered,[2] as summarized in Figure 4.

DEFINITION 4.1 (see Definition 1, point 1 of [24]). *A function space* $V \subseteq H^1(\Omega)$ *is said to gradient embed in* $S \subseteq \left[L^2(\Omega)\right]^d$ *if*

$$\forall \eta \in V \exists u \in S \text{ s.t. } \|u - \nabla\eta\|_{L^2} = 0. \tag{32}$$

DEFINITION 4.2. *A function space* $E \subseteq H^1(\Omega)$ *is said to skew gradient embed in* $S \subseteq \left[L^2(\Omega)\right]^2$ *if*

$$\forall \psi \in E \exists u \in S \text{ s.t. } \left\|u - \nabla^\perp\psi\right\|_{L^2} = 0. \tag{33}$$

These two embedding properties are notably satisfied by the $P1_{DG}$–$P2$ finite element pair in two dimensions, with a conforming triangle mesh, $S = [\mathcal{P}_1]^2$, and $E = V = \mathcal{P}_2 \cap H^1(\Omega)$. It follows that a finite element pair is Ladyzhenskaya–Babuška–Brezzi (LBB) stable [59, 7, 18] if it possesses the gradient embedding property [26, 25]. It further follows that any finite element pair with $E = V$ which possesses both the gradient and skew gradient embedding properties has, when applied to the linearized shallow-water equations on an f-plane, geostrophic modes which are exactly steady [25, 24]. This makes the $P1_{DG}$–$P2$ finite element pair potentially attractive for use in atmosphere and ocean modeling. See [27] for discussions of finite element discretizations with alternative embedding properties, and see [29] for a recent review of discretizations for unstructured mesh ocean modeling.

The existence of both gradient and skew gradient embedding properties can be used to construct a Helmholtz decomposition [24]. In a bounded domain this is not unique and is dependent upon a choice of boundary conditions.

LEMMA 4.3. *Given real Hilbert spaces* $S \subseteq \left[L^2(\Omega)\right]^2$*,* $V \subseteq H^1(\Omega)$*, where* V *gradient and skew gradient embeds in* S*, a Helmholtz decomposition exists:*

$$\forall u \in S \exists! \eta \in V', \psi \in E', h \in S \text{ s.t. } u = \nabla\eta + \nabla^\perp\psi + h, \tag{34}$$

[2] Note that no notational distinction is made between vector and scalar L^2 norms, inner products, and Galerkin L^2 projections.

where $V' = \{\zeta \in V : \langle 1, \zeta \rangle_{L^2} = 0\}$, $E' = E \cap H_0^1(\Omega)$, *and where* h *is weakly harmonic:*

$$\langle \nabla \zeta, h \rangle_{L^2} = 0 \quad \forall \zeta \in V', \tag{35a}$$

$$\left\langle \nabla^\perp \zeta, h \right\rangle_{L^2} = 0 \quad \forall \zeta \in E'. \tag{35b}$$

Proof. The proof is as described in [24], with the difference that here bounded domains are considered. First $\eta \in V'$, $\psi \in E'$ can be defined via

$$\langle \nabla \zeta, \nabla \eta \rangle_{L^2} = \langle \nabla \zeta, u \rangle_{L^2} \quad \forall \zeta \in V', \tag{36a}$$

$$\langle \nabla \zeta, \nabla \psi \rangle_{L^2} = \left\langle \nabla^\perp \zeta, u \right\rangle_{L^2} \quad \forall \zeta \in E', \tag{36b}$$

and existence and uniqueness of η and ψ follow from the Lax–Milgram lemma. Existence of a unique weakly harmonic $h \in S$ follows via orthogonal completion. ∎

Given a $u \in S$, the $\eta \in V'$ defines a weakly divergent component $\nabla \eta \in S$, the $\psi \in E'$ defines a weakly rotational component $\nabla^\perp \psi \in S$, and the $h \in S$ defines a weakly harmonic component. A strong homogeneous Dirichlet boundary condition is applied for ψ, which is the relevant boundary condition when considering a simply connected domain where the rotational component is subject to a no-normal-flow boundary condition on all boundaries. This is, for example, the relevant boundary condition when considering the two-dimensional incompressible Navier–Stokes equations in a simply connected domain with no inflow or outflow.

4.2. Weakly solenoidal L^2 projection. A weakly solenoidal L^2 projection (hereafter "solenoidal projection") can be constructed via the usual constrained optimization problem.

LEMMA 4.4 (solenoidal projection). *Consider real Hilbert spaces* $V_T \subseteq H^1(\Omega)$, $S_D, S_T \subseteq \left[L^2(\Omega)\right]^2$, *where* V_T *gradient embeds in* S_T. *Let*

$$S_{T,1} = \{w_T \in S_T : \langle \nabla \zeta_T, w_T \rangle_{L^2} = 0 \ \forall \zeta_T \in V_T\}.$$

Then for each $u_D \in S_D$ *there exists a unique* $u_T \in S_{T,1}$ *such that*

$$\|u_T - u_D\|_{L^2} \leq \|U_T - u_D\|_{L^2} \quad \forall U_T \in S_{T,1}. \tag{37}$$

This defines a map $\mathcal{I}_{S,1} : S_D \to S_T$ *where*

$$\mathcal{I}_{S,1}(u_D) = \arg \min_{U_T \in S_{T,1}} \|U_T - u_D\|_{L^2}. \tag{38}$$

Proof. The proof is standard and follows from the LBB stability of the finite element pair (see, e.g., [20] and [47, Appendix 5]), which itself follows from gradient embedding [26, 25]. The resulting projection $\mathcal{I}_{S,1} : S_D \to S_T$ is given by

$$\mathcal{I}_{S,1}(u_D) = \Pi(u_D) + \nabla \lambda, \tag{39}$$

where $\lambda \in V_T' = \{\zeta_T \in V_T : \langle 1, \zeta_T \rangle_{L^2} = 0\}$ satisfies

$$\langle \nabla \zeta_T, \nabla \lambda \rangle_{L^2} = -\langle \nabla \zeta_T, \Pi(u_D) \rangle_{L^2} \quad \forall \zeta_T \in V_T'. \tag{40}$$

∎

As described in [20], given the mesh-to-mesh Galerkin projection $\Pi(u_D)$, the L^2 optimal solenoidal projection can be constructed via postprocessing on the target mesh—that is, the solution of the elliptic problem (40) for the Lagrange multiplier λ,

followed by a modification to the Galerkin projection via (39). While the solution of an elliptic problem necessarily leads to an increased cost, if the solenoidal projection is used as part of a mesh adaptive incompressible Navier–Stokes solver, then this is unlikely to be significantly more expensive than a single model timestep.

In order to analyze the error in the solenoidal projection, first note that the Galerkin projection of a weakly rotational function is weakly solenoidal. Note that here, and for the remainder of this section, a boundary condition choice as used in Lemma 4.3 is made.

LEMMA 4.5. *Consider real Hilbert spaces* $V_D, V_T \subseteq H^1(\Omega)$, $S_D, S_T \subseteq \left[L^2(\Omega)\right]^2$, *where* V_D *skew gradient embeds in* S_D *and where* V_T *gradient embeds in* S_T. *Let* $\Pi : S_D \to S_T$ *be a Galerkin projection. Then*

$$\exists \psi_D \in V_D \cap H_0^1(\Omega) \text{ s.t. } u_D = \nabla^\perp \psi_D \implies \langle \nabla \zeta_T, \Pi(u_D) \rangle_{L^2} = 0 \quad \forall \zeta_T \in V_T. \tag{41}$$

Proof. It follows from (2) that

$$\langle w_T, \Pi(u_D) \rangle_{L^2} = \left\langle w_T, \nabla^\perp \psi_D \right\rangle_{L^2} \quad \forall w_T \in S_T, \tag{42}$$

and hence from gradient embedding that

$$\begin{aligned} \langle \nabla \zeta_T, \Pi(u_D) \rangle_{L^2} &= \left\langle \nabla \zeta_T, \nabla^\perp \psi_D \right\rangle_{L^2} \\ &= 0 \quad \forall \zeta_T \in V_T. \end{aligned} \tag{43}$$

∎

This leads to an error bound.

LEMMA 4.6. *Consider real Hilbert spaces* $V_D, V_T \subseteq H^1(\Omega)$, $S_D, S_T \subseteq \left[L^2(\Omega)\right]^2$, *where* V_D *gradient and skew gradient embeds in* S_D *and where* V_T *gradient embeds in* S_T. *Let* $\Pi : S_D \to S_T$ *be a Galerkin projection. Then given a weakly solenoidal* $u_D \in S_D$,

$$\langle \nabla \zeta_D, u_D \rangle_{L^2} = 0 \quad \forall \zeta_D \in V_D, \tag{44}$$

the L^2 *error in the solenoidal projection of* u_D *onto* S_T *is bounded via*

$$\varepsilon_{\mathcal{I}_{S,1}}(u_D) \leq \varepsilon_\Pi(u_D) + \|h_D\|_{L^2}, \tag{45}$$

where $\varepsilon_\Pi(u_D)$ *is the* L^2 *error in the Galerkin projection, and* h_D *is the weakly harmonic component in the Helmholtz decomposition of* u_D *defined using Lemma* 4.3*:*

$$\begin{aligned} \exists! \eta_D \in V_D' = \{\zeta_D \in V_D : \langle 1, \zeta_D \rangle_{L^2} = 0\}, \psi_D \in V_D \cap H_0^1(\Omega), h_D \in S_D \\ \text{s.t. } u_D = \nabla \eta_D + \nabla^\perp \psi_D + h_D. \end{aligned} \tag{46}$$

Proof. It follows immediately from (36) that $\eta_D = 0$. From (40), using linearity of the Galerkin projection, and via Lemma 4.5,

$$\begin{aligned} \|\nabla \lambda\|_{L^2}^2 &= -\langle \nabla \lambda, \Pi(u_D) \rangle_{L^2} \\ &= -\left\langle \nabla \lambda, \Pi\left(\nabla^\perp \psi_D\right) + \Pi(h_D) \right\rangle_{L^2} \\ &= -\langle \nabla \lambda, \Pi(h_D) \rangle_{L^2}, \end{aligned} \tag{47}$$

and hence via the Cauchy–Schwarz inequality and using (9),

$$\begin{aligned} \|\nabla \lambda\|_{L^2} &\leq \|\Pi(h_D)\|_{L^2} \\ &\leq \|h_D\|_{L^2}. \end{aligned} \tag{48}$$

Now using (39) and applying the triangle inequality leads to

$$\begin{aligned}\varepsilon_{\mathcal{I}_{S,1}}(u_D) = \left\|\mathcal{I}_{S,1}\left(u_D\right) - u_D\right\|_{L^2} &\leq \varepsilon_\Pi(u_D) + \|\nabla\lambda\|_{L^2} \\ &\leq \varepsilon_\Pi(u_D) + \|h_D\|_{L^2}. \end{aligned} \tag{49}$$

∎

That is, given a weakly solenoidal donor function, the L^2 error in the solenoidal projection is bounded in terms of the sum of the Galerkin projection L^2 error and the L^2 norm of the weakly harmonic component of the donor function.

In principle the weakly solenoidal donor function u_D in Lemma 4.6 may have a weakly harmonic component of arbitrary magnitude. However, in practice u_D is expected to be a discrete approximation to some exact solenoidal solution, for example, the velocity for the incompressible Navier–Stokes equations. Moreover, in a bounded simply connected domain with no-normal-flow boundary conditions, the existence of nonzero harmonic components is a spurious numerical artifact [24]. This can be expressed in the form of an alternative error bound.

LEMMA 4.7. *Consider a* $\breve{u} \in H\left(\Omega, \mathrm{div}\right)$*, where* $\|\breve{u} - \nabla^\perp\breve{\psi}\|_{L_2} = 0$ *for some* $\breve{\psi} \in H_0^1\left(\Omega\right)$*. Consider function spaces and a weakly solenoidal* $u_D \in S_D$ *as defined in Lemma* 4.6*. Then the* L^2 *error in the solenoidal projection of* $\mathcal{I}_{S,1}\left(u_D\right)$ *onto* S_T *is bounded via*

$$\varepsilon_{\mathcal{I}_{S,1}}(u_D) \leq \varepsilon_\Pi(u_D) + 2\left\|u_D - \breve{u}\right\|_{L^2} + \left\|\breve{\Pi}_{H^1}\left(\breve{\psi}\right) - \breve{\psi}\right\|_{H^1}, \tag{50}$$

where $\breve{\Pi}_{H^1}(\breve{\psi}) \in V_D' = V_D \cap H_0^1\left(\Omega\right)$ *is an* H^1 *projection of* $\breve{\psi}$*, defined via*

$$\left\langle \nabla\zeta_D, \nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right)\right\rangle_{L^2} = \left\langle \nabla\zeta_D, \nabla\breve{\psi}\right\rangle_{L^2} \quad \forall \zeta_D \in V_D'. \tag{51}$$

Proof. The Helmholtz decomposition of u_D and the triangle inequality lead to

$$\begin{aligned}\|h_D\|_{L^2} &= \left\|u_D - \nabla^\perp\psi_D\right\|_{L^2} \\ &\leq \left\|u_D - \breve{u}\right\|_{L^2} + \left\|\nabla\psi_D - \nabla\breve{\psi}\right\|_{L^2}. \end{aligned} \tag{52}$$

Now note that $\nabla\breve{\Pi}_{H^1}(\breve{\psi})$ is defined via an orthogonal projection,

$$\begin{aligned}\left\|\nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right) - \nabla\breve{\psi}\right\|_{L^2}^2 &= \left\|\nabla\breve{\psi}\right\|_{L^2}^2 + \left\|\nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right)\right\|_{L^2}^2 - 2\left\langle \nabla\breve{\psi}, \nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right)\right\rangle_{L^2} \\ &= \left\|\nabla\breve{\psi}\right\|_{L^2}^2 - \left\|\nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right)\right\|_{L^2}^2, \end{aligned} \tag{53}$$

where the second line follows using (51). Hence,

$$\begin{aligned}\left\|\nabla\psi_D - \nabla\breve{\psi}\right\|_{L^2}^2 &= \|\nabla\psi_D\|_{L^2}^2 + \left\|\nabla\breve{\psi}\right\|_{L^2}^2 - 2\left\langle \nabla\psi_D, \nabla\breve{\psi}\right\rangle_{L^2} \\ &= \|\nabla\psi_D\|_{L^2}^2 + \left\|\nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right) - \nabla\breve{\psi}\right\|_{L^2}^2 \\ &\quad + \left\|\nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right)\right\|_{L^2}^2 - 2\left\langle \nabla\psi_D, \nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right)\right\rangle_{L^2} \\ &= \left\|\nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right) - \nabla\breve{\psi}\right\|_{L^2}^2 + \left\|\nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right) - \nabla\psi_D\right\|_{L^2}^2, \end{aligned} \tag{54}$$

where the second line follows using (51) and (53). It follows from this and (52) that

$$\|h_D\|_{L^2} \leq \|u_D - \breve{u}\|_{L^2} + \left\|\nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right) - \nabla\breve{\psi}\right\|_{L^2} + \left\|\nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right) - \nabla\psi_D\right\|_{L^2}. \tag{55}$$

Now from (36b) and (51),

$$\left\langle \nabla\zeta_D, \nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right) - \nabla\psi_D\right\rangle_{L^2} = \left\langle \nabla\zeta_D, -\breve{u}^{\perp} + u_D^{\perp}\right\rangle_{L^2} \quad \forall \zeta_D \in V_D', \tag{56}$$

and hence via the Cauchy–Schwarz inequality,

$$\left\|\nabla\breve{\Pi}_{H^1}\left(\breve{\psi}\right) - \nabla\psi_D\right\|_{L^2} \leq \|u_D - \breve{u}\|_{L^2}, \tag{57}$$

leading to

$$\|h_D\|_{L^2} \leq 2\|u_D - \breve{u}\|_{L^2} + \left\|\breve{\Pi}_{H^1}\left(\breve{\psi}\right) - \breve{\psi}\right\|_{H^1}. \tag{58}$$

With Lemma 4.6 this leads to (50). ∎

Hence the L^2 error in the mesh-to-mesh solenoidal projection is bounded in terms of the L^2 mesh-to-mesh Galerkin projection error, the L^2 discretization error in u_D (the discrete "velocity" in the discrete donor "velocity" space), and the H^1 error in the H^1 projection of the exact stream function onto the discrete donor "stream function" space. For the $P1_{DG}$–$P2$ finite element pair, the third term in (50) converges asymptotically at second order in the donor mesh element size given a sufficiently regular $\breve{\psi}$ and nonpathological meshes (see [22, section 3-4-2]), and the asymptotic convergence rate of the second term in (50) is dependent upon the details of the discretization used to yield u_D.

4.3. L^2 norm preserving weakly solenoidal interpolation. The solenoidal and L^2 norm preserving interpolants can now be combined via the introduction of a nonlinear constraint.

LEMMA 4.8. *Consider real Hilbert spaces $V_T \subseteq H^1(\Omega)$ and $S_D, S_T \subseteq \left[L^2(\Omega)\right]^2$, where V_T gradient embeds in S_T. Let $\mathcal{I}_{S,1} : S_D \to S_T$ be a solenoidal projection. For each $u_D \in S_{D,2} = \left\{w_D \in S_D : \|\mathcal{I}_{S,1}(w_D)\|_{L^2} \neq 0\right\}$ there exists a unique $u_T \in S_{T,2,u_D} = \{w_T \in S_T : \langle \nabla\zeta_T, w_T\rangle_{L^2} = 0 \ \forall \zeta_T \in V_T, \|w_T\|_{L^2} = \|u_D\|_{L^2}\}$ such that*

$$\|u_T - u_D\|_{L^2} \leq \|U_T - u_D\|_{L^2} \quad \forall\ U_T \in S_{T,2,u_D}. \tag{59}$$

This defines a map $\mathcal{I}_{S,2} : S_{D,2} \to S_T$ where

$$\mathcal{I}_{S,2}(u_D) = \arg\min_{U_T \in S_{T,2,u_D}} \|U_T - u_D\|_{L^2}. \tag{60}$$

Proof. Given a $u_D \in S_{D,2}$, define a functional $J_{S,2} : S_{D,2} \to \mathbb{R}$ where

$$J_{S,2}(U_T) = \frac{1}{2}\|U_T - u_D\|_{L^2}^2. \tag{61}$$

Noting that $S_{T,2,u_D} \subseteq S_{T,1}$, define the Lagrange constrained functional $\hat{J}_{S,2} : S_{T,1} \times \mathbb{R} \to \mathbb{R}$ where

$$\hat{J}_{S,2}(U_T, \Lambda) = \frac{1}{2}\|U_T - u_D\|_{L^2}^2 + \frac{\Lambda}{2}\left[\|U_T\|_{L^2}^2 - \|u_D\|_{L^2}^2\right], \tag{62}$$

and seek critical points $(u_T, \lambda) \in S_T \times \mathbb{R}$ at which the Gâteaux derivative vanishes:

$$\mathrm{d}\hat{J}_{S,2}\left(u_T, \lambda; w_T, 0\right) = \left\langle w_T, (1+\lambda)\, u_T - u_D \right\rangle_{L^2} = 0 \quad \forall w_T \in S_{T,1}, \tag{63a}$$

$$\mathrm{d}\hat{J}_{S,2}\left(u_T, \lambda; 0, 1\right) = \frac{1}{2}\left\|u_T\right\|_{L^2}^2 - \frac{1}{2}\left\|u_D\right\|_{L^2}^2 = 0. \tag{63b}$$

By inspection, $(1+\lambda)\, u_T = \mathcal{I}_{S,1}\left(u_D\right)$. The proof then proceeds similarly to that for Lemma 3.1, and minimization of $J_{S,2}$ defines a function $\mathcal{I}_{S,2} : S_{D,2} \to S_T$ where

$$\mathcal{I}_{S,2}\left(u_D\right) = \frac{\left\|u_D\right\|_{L^2}}{\left\|\mathcal{I}_{S,1}\left(u_D\right)\right\|_{L^2}} \mathcal{I}_{S,1}\left(u_D\right). \tag{64}$$ ∎

This is simply a combination of the solenoidal projection $\mathcal{I}_{S,1}$ and the L^2 norm preserving projection $\mathcal{I}_1$. A relation between the error associated with $\mathcal{I}_{S,2}$ and the solenoidal projection error also follows.

LEMMA 4.9. *The L^2 norm preserving solenoidal interpolant has an L^2 error $\varepsilon_{\mathcal{I}_{S,2}}(u_D)$ which can be bounded in terms of the solenoidal projection L^2 error, $\varepsilon_{\mathcal{I}_{S,1}}(u_D)$,*

$$\varepsilon_{\mathcal{I}_{S,2}}(u_D) \leq \sqrt{2}\varepsilon_{\mathcal{I}_{S,1}}(u_D), \tag{65a}$$

and which has an asymptotic expansion

$$\varepsilon_{\mathcal{I}_{S,2}}(u_D) = \varepsilon_{\mathcal{I}_{S,1}}(u_D) + \frac{1}{8}\frac{\varepsilon_{\mathcal{I}_{S,1}}(u_D)^3}{\left\|u_D\right\|_{L^2}^2} + \mathcal{O}\left(\frac{\varepsilon_{\mathcal{I}_{S,1}}(\phi_D)^5}{\left\|u_D\right\|_{L^2}^4}\right) \text{ as } \frac{\varepsilon_{\mathcal{I}_{S,1}}(u_D)^5}{\left\|u_D\right\|_{L^2}^4} \to 0. \tag{65b}$$

Proof. $\mathcal{I}_{S,1}$ is an orthogonal projection,

$$\begin{aligned}
\varepsilon_{\mathcal{I}_{S,1}}(u_D)^2 &= \left\|\mathcal{I}_{S,1}\left(u_D\right)\right\|_{L^2}^2 + \left\|u_D\right\|_{L^2}^2 - 2\left\langle \mathcal{I}_{S,1}\left(u_D\right), \Pi\left(u_D\right)\right\rangle_{L^2} \\
&= \left\|\mathcal{I}_{S,1}\left(u_D\right)\right\|_{L^2}^2 + \left\|u_D\right\|_{L^2}^2 - 2\left\langle \mathcal{I}_{S,1}\left(u_D\right), \mathcal{I}_{S,1}\left(u_D\right) - \nabla\lambda\right\rangle_{L^2} \\
&= \left\|u_D\right\|_{L^2}^2 - \left\|\mathcal{I}_{S,1}\left(u_D\right)\right\|_{L^2}^2, \qquad (66)
\end{aligned}$$

where the final line follows as $\mathcal{I}_{S,1}\left(u_D\right)$ is weakly solenoidal. The proof proceeds similarly to that for Lemma 3.2. ∎

Hence the L^2 norm preserving solenoidal interpolant is a higher order modification to the solenoidal projection.

5. Numerical example: Two-dimensional lock-exchange. The lock-exchange is a widely studied laboratory-scale fluid dynamics problem (see, e.g., [12, 79, 51]). It is an excellent test case for numerical models, as the configuration is simple but the flow produced is complex, as shown in Figure 5 (with associated meshes shown in Figure 6). Initially, a vertical barrier separates a flat-bottomed tank into two sections, with one section filled with relatively dense fluid and the other with relatively light fluid. Once the barrier is removed, the system enters the "propagation stage," where denser fluid collapses under the lighter, forming two gravity currents that propagate in opposite directions, one above the other, along the tank. Once the gravity current fronts reach the end walls, the system enters the "oscillatory stage," with the fluid "sloshing" back and forth across the tank. Eventually the motion will subside. Shear instabilities, Kelvin–Helmholtz billows, internal waves, and interaction with the end walls can all lead to enhanced turbulence and diapycnal mixing between the fluids of different densities. The numerical representation of this system is challenging, with processes and energy transfers occurring across multiple scales. Hence the system is potentially sensitive to numerical artifacts introduced by mesh-to-mesh interpolation.

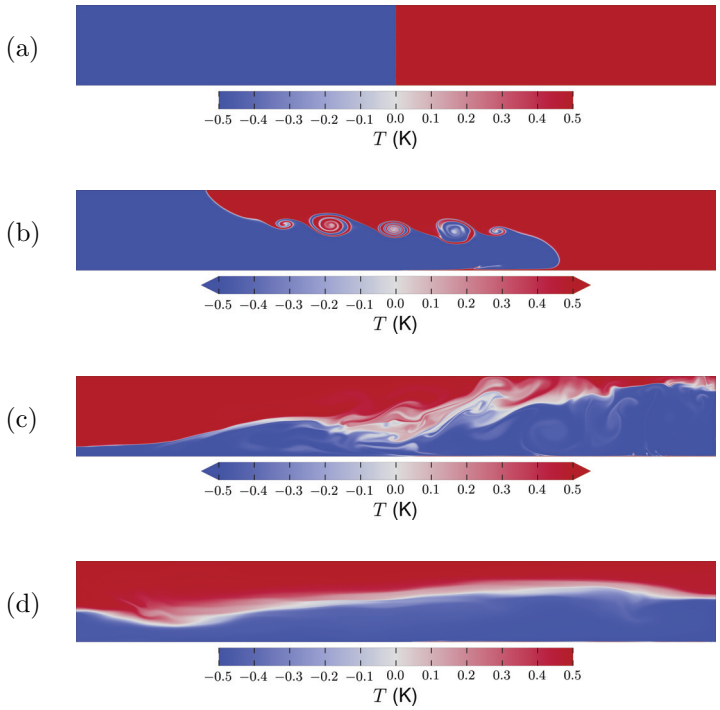


FIG. 5. *Snapshots of the temperature for the simulation of the two-dimensional lock-exchange using configuration L2S_L2P (see Table 3). Note that the color scales are not extended to include numerical over- and under-shoots.* (a) *Time* $t/T_b = 0$*, initial condition.* (b) *Time* $t/T_b = 0.7858$*, formation of Kelvin–Helmholtz billows.* (c) *Time* $t/T_b = 3.931$*, flow after the gravity currents reach the end walls, with the fluid "sloshing" back and forth across the tank.* (d) *Time* $t/T_b = 14.94$.

5.1. Configuration. The two-dimensional lock-exchange is simulated with Fluidity (see, e.g., [73, 30]) following [55] (see also [54, 53]). The system is governed by the Navier–Stokes equations under the Boussinesq approximation, a linear equation of state, and the thermal advection-diffusion equation. The system is initially at rest, with dense cold water, with temperature[3] $T = -0.5$ K, filling one half of the domain and light warm water, with temperature $T = +0.5$ K, filling the other half. A free-slip, no-normal-flow boundary condition is applied to the end walls and the top boundary, and a no-slip boundary condition is applied to the bottom boundary. Physical parameters are listed in Table 2.

A $P1_{DG}$–$P2$ discretization [26, 25] and the projection method of [39] are used for the velocity and pressure. The time discretization uses a twice Picard iterated approximation to a Crank–Nicolson discretization for the coupled velocity and pressure system (see [39, section 2.c]). Upwinding is applied for interfacial advective fluxes, the scheme described in [11] is used for the viscous term, and all velocity boundary conditions are applied weakly. A $P1$ temperature is defined, and the thermal advection equation is discretized via a node-centered control-volume advection scheme [60], with the Sweby limiter [82], conservative advection, and zero explicit diffusion. The temperature equation is discretized in time with subcycling as described in [4, section 3.4.2] (with $\theta = \frac{1}{2}$, $\theta_p = 1$, and three subcycle iterations). Note that the velocity and

[3] More precisely, the temperature perturbation relative to a background temperature T_0.

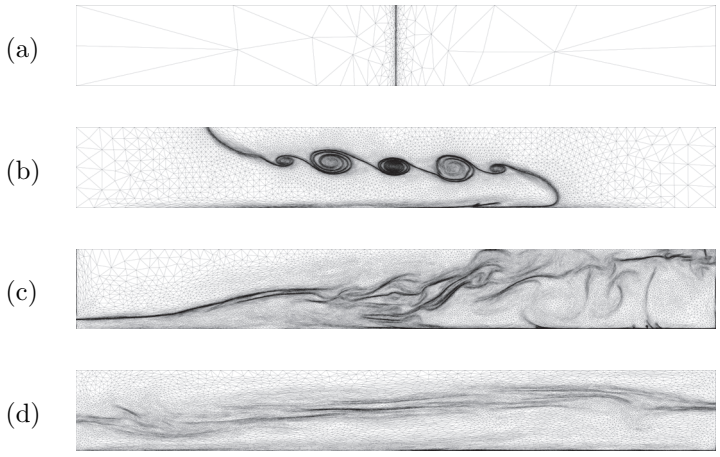


FIG. 6. *Snapshots of the mesh for the simulation of the two-dimensional lock-exchange using configuration L2S_L2P (see Table 3). Times are as for Figure 5.*

TABLE 2
Physical parameters for the two-dimensional lock-exchange per [55].

Quantity	Symbol	Value
Domain length	L	0.8 m
Domain height	$H = 2h$	0.1 m
Gravitational acceleration	g	10 $\mathrm{m}^2\ \mathrm{s}^{-1}$
Thermal expansion coefficient	α	$10^{-3}\ \mathrm{K}^{-1}$
Initial temperature difference	ΔT	1 K
Kinematic viscosity	ν	$10^{-6}\ \mathrm{m}^2\ \mathrm{s}^{-1}$
Buoyancy velocity	$u_b = \sqrt{g\alpha\Delta T H}$	$\sqrt{10^{-3}}\ \mathrm{m\ s}^{-1} = 0.03162\ \mathrm{m\ s}^{-1}$
Buoyancy period	$T_b = \frac{2\pi H}{u_b}$	$2\pi\sqrt{10}$ s = 19.87 s
Grashof number	$\mathrm{Gr} = \frac{h^3 g\alpha\Delta T}{\nu^2}$	1.25×10^6
Reynolds number	$\mathrm{Re} = \frac{u_b h}{\nu}$	$5\sqrt{10^5} = 1581$

temperature discretizations do not use a consistent discretization for the divergence operator [88]. This issue is addressed by projecting the $P1_{DG}$ velocity onto the space of functions with zero control volume divergence and by using this projected velocity to advect the temperature.

Two adaptive mesh settings are considered, corresponding to the M_∞-const and M_2-mid cases of [55]. These adapt the mesh based upon a Hessian-based metric derived through consideration of the interpolation error under the L_∞ [71] and L_2 norm [42, 21, 63], respectively (see equation 9 of [55]). The M_∞-const configuration preferentially concentrates mesh resolution in the regions of highest solution curvature. By contrast the M_2-mid configuration spreads the resolution into regions of relatively lower curvature, and this is found in [55] to lead to lower numerically induced mixing.

The metrics require approximations for function Hessians, which are more easily constructed for solutions in H^1, and hence the Hessian-based metric associated with the velocity is computed indirectly from the Galerkin L^2 projection of the $P1_{DG}$ velocity components onto a $P1$ function space. The three metrics are superimposed as described in [71], and a gradation algorithm from [16] is applied. Additional postpro-

cessing of the metric is applied, including the specification of a minimum edge length of 10^{-4} m and a maximum edge length of 0.5 m. The mesh optimization itself is performed using a modified version of the Ani2D-MBA package from Ani2D [85].

The kinetic energy of the system is

$$E_k = \frac{1}{2}\rho_0 \int_\Omega u \cdot u, \tag{67}$$

where ρ_0 is a constant reference density and u is the velocity. Hence an L^2 norm preserving interpolant for the velocity (in particular, the L^2 norm preserving solenoidal interpolant $\mathcal{I}_{S,2}$ in (64)) conserves kinetic energy. The potential energy of the system is, subject to an appropriate choice of potential energy zero point,

$$E_p = -\rho_0 \int_\Omega g\alpha T z dz, \tag{68}$$

where z is the vertical coordinate. Since $g\alpha$ is a constant it follows that Galerkin L^2 projection, applied to the $P1$ temperature field, conserves potential energy. An L^2 norm preserving conservative interpolant which also conserves the potential energy can be constructed using $X = \text{span}(\{1, z\})$ in the definition of Π_X in (28).

The tests use five mesh-to-mesh interpolants: consistent interpolation, Galerkin L^2 projection, the L^2 optimal L^2 norm conservative and potential energy conserving interpolant for temperature, the L^2 solenoidal projection for the velocity, and the L^2 optimal L^2 norm preserving solenoidal interpolant. The latter three interpolants are implemented using the FEniCS system [56, 57, 2, 62, 69, 61, 3] via the Fluidity Python interfaces [67]. As an example, the implementation of the L^2 norm preserving solenoidal interpolant is shown in Figure 7. The FEniCS system is also used to compute a number of diagnostics. Supermeshing is performed using the local supermeshing approach of [36], with triangle intersections performed using the Wild Magic 4 library [31]. Although not applied here (but see, e.g., [36]) Fluidity can additionally perform supermeshing of three-dimensional tetrahedra using the "plane-at-a-time clipping" algorithm described in [31, section 2.4.3]. Consistent interpolation defines an interpolant $\mathcal{I}_C : V_D \to V_T$, where $V_D, V_T \subseteq C^0(\Omega)$, by setting the interpolated function to be equal to the donor function at a set of $\dim V_T$ distinct collocation points, which are here chosen to be the target mesh vertices. The considered combination of interpolants is listed in Table 3.

The parallel adaptivity algorithm implemented in Fluidity performs several iterations of mesh optimization, interpolation, and load-rebalancing, in order to permit optimization of shared "halo" regions while retaining a conforming mesh [44, 80]. By default, three such iterations are performed, and hence data are interpolated three times. In order to avoid the associated increased interpolation errors and provide a stricter comparison of the differing combinations of interpolants, the simulations are here conducted in serial. Note that the interfacing between Fluidity and FEniCS is also not currently parallelized.

Simulations are integrated for a total simulation time of 1000 s = 50.33 T_b, with a timestep size of $\Delta t = 0.0125$ s and mesh adaptivity applied every 0.125 s. Linear systems are solved with iterative Krylov methods using PETSc [10, 8, 9], with the pressure system preconditioned using an algebraic multigrid method [84, 58] and the elliptic problem for the solenoidal interpolant preconditioned using BoomerAMG [52, 33]. All tests use Fluidity version 4.1.11, PETSc version 3.4.5, and FEniCS version 1.4.

```
# Solve for the Lagrange multiplier
lam = Function(space_p)
solve(inner(grad(test_p), grad(trial_p)) * dx == -inner(grad(test_p), u) * dx,
     lam,
  DirichletBC(space_p, 0.0, "fabs(x[0]) < DOLFIN_EPS && fabs(x[1]) < DOLFIN_EPS"
       ,
    method = "pointwise"),
  solver_parameters = {"linear_solver":"cg", "preconditioner":"amg", "symmetric"
       :True,
    "krylov_solver":{"relative_tolerance":1.0e-12, "absolute_tolerance":1.0e
         -16}})
# Compute the solenoidal projection
solve(inner(test_u, trial_u) * dx == inner(test_u, u[0] + grad(lam)[0]) * dx, u
     [0],
  solver_parameters = {"linear_solver":"lu"})
solve(inner(test_u, trial_u) * dx == inner(test_u, u[1] + grad(lam)[1]) * dx, u
     [1],
  solver_parameters = {"linear_solver":"lu"})
# Scale to ensure L^2 norm preservation
alpha = sqrt(state["u_sq_int"] / assemble(dot(u, u) * dx))
u[0].vector().set_local(alpha * u[0].vector().array())
u[0].vector().apply("insert")
u[1].vector().set_local(alpha * u[1].vector().array())
u[1].vector().apply("insert")
```

FIG. 7. *Section of Python code implementing the L^2 norm preserving solenoidal interpolant using the FEniCS system (edited for formatting and to add explanatory comments).* `u` *is the Galerkin L^2 projection of the velocity, and* `state["u_sq_int"]` *is the squared L^2 norm of the donor velocity.* `test_p` *and* `test_u` *represent test functions in V_T and S_T, respectively, and* `trial_p` *and* `trial_u` *represent corresponding trial functions. Lines 2–7 correspond to equation (40), lines 9–12 to equation (39), and lines 14–18 to equation (64). Note that here V_T' is defined via the application of a single node boundary condition rather than by the zero integral constraint applied in Lemma 4.4.*

TABLE 3

Mesh-to-mesh interpolant configurations for the two-dimensional lock-exchange simulations. Π refers to Galerkin L^2 projection (section 2.1), $\mathcal{I}_C$ to consistent interpolation, $\mathcal{I}_X$ to the L^2 norm preserving interpolant which additionally preserves inner products with functions in X (section 3.2), $\mathcal{I}_{S,1}$ to the L^2 solenoidal projection (section 4.2), and $\mathcal{I}_{S,2}$ to the L^2 norm preserving solenoidal interpolant (section 4.3).

Configuration	Velocity interpolant	Temperature interpolant	Pressure interpolant
GP_CI	Π	$\mathcal{I}_C$	Π
GP_GP	Π	Π	Π
GP_L2P	Π	$\mathcal{I}_X$ with $X = \text{span}(\{1, z\})$	Π
S_L2P	$\mathcal{I}_S$	$\mathcal{I}_X$ with $X = \text{span}(\{1, z\})$	Π
L2S_L2P	$\mathcal{I}_{S,2}$	$\mathcal{I}_X$ with $X = \text{span}(\{1, z\})$	Π

A closely related configuration, using a stabilized $P1$–$P1$ discretization for the velocity and pressure [73], is described in [55]. Simulations using this latter configuration were shown to be comparable to a high-resolution fixed mesh benchmark with two orders of magnitude more degrees of freedom. A similar configuration using a $P1_{DG}$–$P2$ discretization for velocity and pressure is described in [36].

5.2. Computational mesh. The degrees of freedom in the temperature space are shown in Figure 8. The number of degrees of freedom increases rapidly during the initial propagation stage and the formation of the gravity current. It then evolves irregularly during the early oscillatory stage, as the mesh adapts to the complex flow

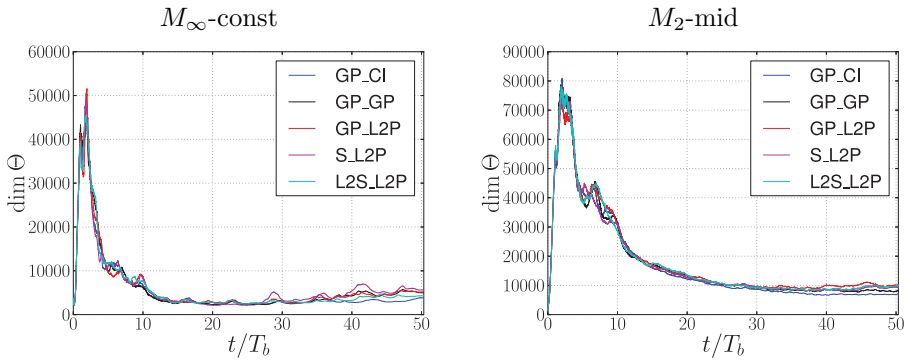


FIG. 8. *Degrees of freedom in the temperature space* Θ *for the* M_∞*-const case (left) and the* M_2*-mid case (right).*

after the gravity currents hit the end walls. At later times the number of degrees of freedom in each space decreases to, and then varies around, a lower value, as the flow subsides towards a steady-state solution.

5.3. Conservation and boundedness. The mean temperature change and temperature variance are shown in Figure 9. All configurations except for the GP_CI configuration conserve the mean temperature, except for errors associated with numerical round-off and linear solver tolerances. In the M_∞-const configuration the GP_L2P configuration has the lowest dissipation of temperature variance, followed by S_L2P and L2S_L2P. That is, the switch to a solenoidal interpolant for the velocity leads to an increased dissipation of temperature variance, and this is further increased by the addition of kinetic energy conservation. A similar picture is observed in the M_2-mid configuration, where now the L2S_L2P configuration has greater dissipation of temperature variance than the GP_GP configuration.

Since Galerkin L^2 projection dissipates in the L^2 norm, the GP_L2P configuration loses kinetic energy in each interpolation, and hence it is perhaps not surprising that this configuration leads to weaker mixing than the L2S_L2P configuration, which conserves kinetic energy in each interpolation and hence has more energy available for tracer mixing. Increased mixing is also observed when switching from Galerkin L^2 projection for the velocity to solenoidal projection.

The discretization of the thermal advection-diffusion equation does not preserve the temperature bounds. Moreover, Galerkin L^2 projection does not preserve the bounds of an interpolated function, and an L^2 norm conserving interpolant magnifies any resulting numerical over- and under-shoots. Consistent interpolation of a linear function, by contrast, does not introduce spurious over- or under-shoots. Figure 10 shows the maximum and minimum temperatures for the GP_CI, GP_GP and L2S_L2P configurations. The latter configuration, using an L^2 norm preserving interpolant for the temperature, has over-shoot magnitudes similar to those of the GP_GP case, although the over-shoot affects a greater fraction of the domain at early times (not shown). However, numerical dissipation overcomes this effect at later times.

5.4. Energetics and numerically induced mixing. The dominant energy exchange in the lock-exchange is between the potential and kinetic energy. As the fluid sloshes back and forth, the kinetic and potential energy oscillate. However, the sum of the potential and kinetic energy decreases over time as energy is lost to mix-

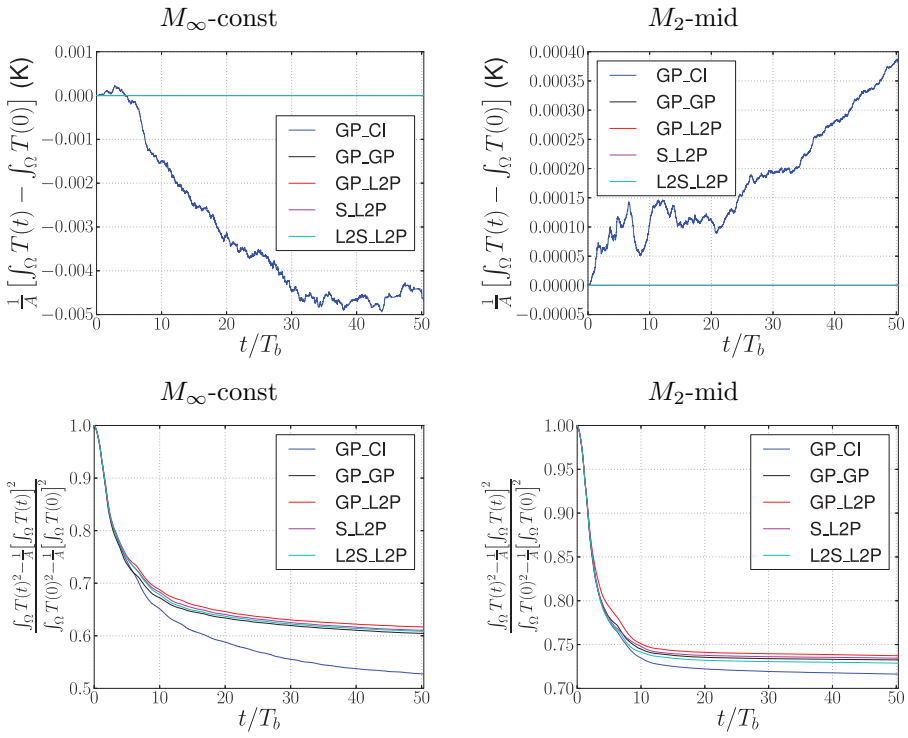


FIG. 9. *Mean temperature change (upper row) and mean temperature variance (lower row) for the M_∞-const case (left) and M_2-mid case (right). The L2S_L2P case overlays the GP_GP, GP_L2P, and S_L2P configurations in the upper figures.*

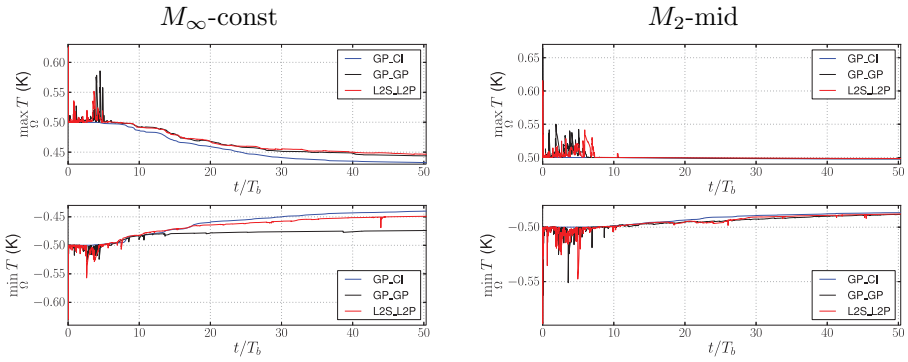


FIG. 10. *Maximum (above) and minimum (below) temperatures for the M_∞-const case (left) and M_2-mid case (right).*

ing. The background potential energy, which is the potential energy of the minimum energy state (or reference state) that can be obtained by adiabatic redistribution of the system, can be used to quantify the energy loss [91, 90]. Most crucially, for a mixing and closed system, changes to the reference state caused by diapycnal mixing correspond to increases in the background potential energy [91]. While the potential and kinetic energy oscillate, the background potential energy is, for a mixing system,

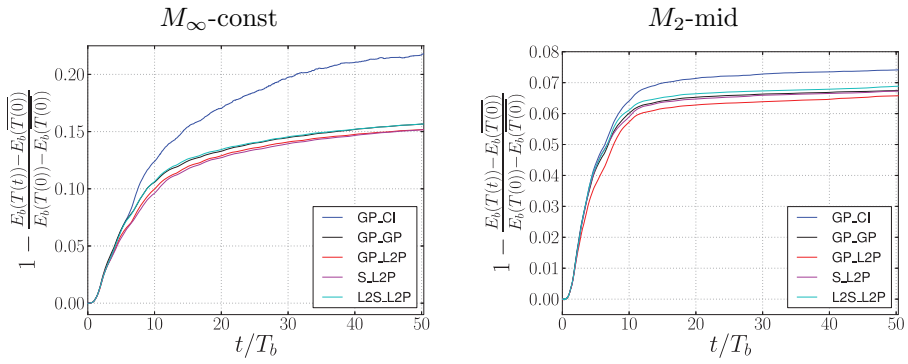


FIG. 11. *Background potential energy for the two-dimensional lock-exchange simulations. Here the abscissa takes a value of zero for the initial condition and a value of one for a completely homogenized state, with temperature equal to the mean initial temperature everywhere.*

a monotonically increasing function that allows the loss of energy to diapycnal mixing to be clearly evaluated. Here, the diffusion term in the thermal advection-diffusion equation is neglected. Any diapycnal mixing that occurs, therefore, is spurious and can be attributed to the numerics. It is preferable to minimize this effect, and hence a smaller increase in the background potential energy over time indicates an improved simulation performance.

The background potential energy, subject to an appropriate choice of potential energy zero point, is

$$E_b = -L\rho_0 \int_0^H g\alpha T_* z_* dz_*, \tag{69}$$

where z_* is the vertical coordinate of fluid with temperature T_* in the reference state. z_* is defined via

$$z_*(T_*) = H \frac{\iint_\Omega \mathcal{H}(x,z,T_*)\, dxdz}{\iint_\Omega 1 dxdz}, \tag{70a}$$

$$\mathcal{H}(x,z,T_*) = \begin{cases} 0 & \text{if } T(x,z) > T_*, \\ 1 & \text{otherwise,} \end{cases} \tag{70b}$$

where T is the temperature. Here z_* is calculated using the method of [83], yielding 10,001 values of z_* at uniformly spaced values of T_* in the interval $[\min_\Omega T, \max_\Omega T]$. E_b/ρ_0 is computed from these values using trapezoidal rule quadrature, with the diagnostic computed every 0.625 s.

Figure 11 shows the background potential energy. The GP_CI configuration leads to significantly larger mixing, particularly for the M_∞-const case (as compared to the alternative configurations in each case). The GP_L2P configuration leads to reduced mixing. In the M_2-mid case the S_L2P configuration has greater mixing than the GP_L2P configuration, and the L2S_L2P configuration has further increased mixing. In the M_∞-const case the L2S_L2P and GP_GP configurations have similar levels of numerically induced mixing, while in the M_2-mid case the L2S_L2P configuration has greater numerically induced mixing than the GP_GP configuration.

5.5. Computational cost. The additional cost of the interpolants was measured using the `MPI_Wtime` MPI function, with timings taken over the first 190

timestep-adapt-interpolate cycles for the M_2-mid L2S_L2P configuration. The timings were measured on a machine with a 3.20 GHz Intel Core i5-3470 processor. The mesh had an average of 27,844 vertices and 55,436 elements (excluding the optimized mesh generated in the final adapt). The total runtime was 39,735 s. The model timestepping (including some diagnostics) accounted for 88.3% of the model runtime. Mesh-to-mesh Galerkin projection via supermeshing accounted for 3.5% of the runtime, and the mesh optimization using the modified version of Ani2D-MBA accounted for 0.3% of runtime. The additional cost of interfacing with FEniCS and performing the postprocessing steps to construct the modified velocity and temperature interpolants (including some diagnostics) was 6.2% of runtime.

6. Conclusions. Once the Galerkin L^2 projection between function spaces defined on different meshes is available, more specialized constrained interpolants which are both L^2 optimal and preserve additional properties can be formulated. This principle was applied to construct interpolants which are L^2 optimal and preserve the L^2 norm, and additionally preserve either the function integral (for scalar functions) or preserve incompressibility (for function spaces with gradient and skew-gradient embedding properties).

Specifically an L^2 optimal L^2 norm preserving interpolant was derived. This takes the form of a scaling of the result following Galerkin L^2 projection. The L^2 optimal, globally conservative, and L^2 norm preserving interpolant was derived similarly. These two interpolants are higher order modifications to Galerkin L^2 projection, and introduce an additional error which is asymptotically third order in the Galerkin L^2 projection error (see Lemmas 3.2 and 3.4).

In addition, interpolants for weakly solenoidal vector-valued functions were derived. For function spaces which observe gradient and skew-gradient embedding properties, such as for the $P1_{DG}$–$P2$ finite element pair, an L^2 optimal solenoidal interpolant was studied. This yields the standard orthogonal projection onto a weakly solenoidal space, with the difference here that mesh-to-mesh Galerkin L^2 projection is applied in order to yield an optimal mesh-to-mesh solenoidal projection. For the function spaces considered, the projection error was bounded in terms of the Galerkin L^2 projection error and the magnitude of the weakly harmonic component of the donor function. For simulations of incompressible flow in a bounded simply connected domain with no inflow or outflow, this latter contribution was itself related to the discretization error in the donor function. The mesh-to-mesh solenoidal projection was combined with the norm preserving interpolant to yield an L^2 optimal L^2 norm preserving solenoidal interpolant.

These interpolants were tested via simulations of the two-dimensional lock-exchange. The GP_L2P configuration, using an L^2 norm preserving, globally conservative, and potential energy conserving interpolant for the temperature, was found to have marginally lower numerically induced mixing than the GP_GP configuration, which used Galerkin L^2 projection alone. The use of a solenoidal interpolant for the velocity in one case led to increased numerically induced mixing, and the L^2 norm conserving solenoidal interpolant for the velocity led to further increased mixing. Hence here some competition was observed between interpolants which preserve temperature variance and lower mixing, and interpolants which preserve discrete properties for the velocity, which in some cases were here found to increase mixing.

In [55] it is concluded that the mesh quality metric is critical to the simulation quality for the two-dimensional lock-exchange problem. However, the effect of the mesh-to-mesh interpolant, while smaller, has here been found to have a measurable

effect on the simulation quality in this complex numerical test case.

Constrained L^2 optimal interpolants can be formulated very generally via the introduction of appropriate Lagrange constraints [20]. Moreover, as a consequence of (2), if the constrained functional involves only L^2 inner products with the donor data, then constrained interpolants can always be constructed using a single Galerkin L^2 projection of the donor function onto an appropriate target space defined on the target mesh, and then via additional postprocessing (but without further multimesh integration). Similarly, if the constrained functional involves only H^1 semi-inner products with the donor data, then the constrained interpolant can always be constructed using a single Galerkin L^2 projection of the function *gradient* onto an appropriate target space defined on the target mesh. Hence the availability of mesh-to-mesh Galerkin L^2 projection enables the straightforward implementation of a large class of constrained interpolants. This property has been applied here in order to formulate L^2 norm preserving and incompressibility preserving interpolants. However, the approach can be generalized to enable the development and implementation of more advanced L^2 optimal interpolants which preserve essential discrete properties.

Acknowledgments. The solenoidal projection was previously implemented in Fluidity by P. E. Farrell and C. R. Wilson. JRM acknowledges useful discussions with J. W. Pearson. Data from the supermesh example in section 2.1, and from the numerical example in section 5, are available through Edinburgh DataShare (https://doi.org/10.7488/ds/1362). The Fluidity code, used in this paper, is available from http://fluidityproject.github.io/. The FEniCS system, used in this paper, is available from http://fenicsproject.org/. We would like to thank the two anonymous reviewers for comments which significantly improved this paper.

REFERENCES

[1] A. Adcroft and D. Marshall, *How slippery are piecewise-constant coastlines in numerical ocean models?*, Tellus A, 50 (1998), pp. 95–108.

[2] M. S. Alnæs, A. Logg, K.-A. Mardal, O. Skavhaug, and H. P. Langtangen, *Unified framework for finite element assembly*, Int. J. Comput. Sci. Engrg., 4 (2009), pp. 231–244.

[3] M. S. Alnæs, A. Logg, K. B. Ølgaard, M. E. Rognes, and G. N. Wells, *Unified form language: A domain-specific language for weak formulations of partial differential equations*, ACM Trans. Math. Software, 40 (2014), 9.

[4] Applied Modelling and Computation Group (AMCG), *Fluidity Manual*, version 4.1.12, Department of Earth Science and Engineering, Imperial College London, London, UK, 2015.

[5] A. Arakawa, *Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow. Part* I, J. Comput. Phys., 1 (1966), pp. 119–143.

[6] D. N. Arnold, R. S. Falk, and R. Winther, *Finite element exterior calculus: From Hodge theory to numerical stability*, Bull. Amer. Math. Soc., 47 (2010), pp. 281–354.

[7] I. Babuška, *The finite element method with Lagrangian multipliers*, Numer. Math., 20 (1973), pp. 179–192.

[8] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, S. Zampini, and H. Zhang, *PETSc Users Manual*, Tech. Report ANL-95/11, Revision 3.6, Argonne National Laboratory, Lemont, IL, 2015.

[9] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, S. Zampini, and H. Zhang, *PETSc Web Page*, http://www.mcs.anl.gov/petsc. (Accessed 04-25-17.)

[10] S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith, *Efficient management of parallelism in object-oriented numerical software libraries*, in Modern Software Tools in Scientific Computing, E. Arge, A. M. Bruaset, and H. P. Langtangen, eds., Birkhäuser Boston, Inc., Boston, 1997, pp. 163–202.

[11] F. Bassi and S. Rebay, *A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations*, J. Comput. Phys., 131 (1997), pp. 267–279.

[12] T. B. Benjamin, *Gravity currents and related phenomena*, J. Fluid Mech., 31 (1968), pp. 209–248.

[13] P.-E. Bernard, N. Chevaugeon, V. Legat, E. Deleersnijder, and J.-F. Remacle, *High-order h-adaptive discontinuous Galerkin methods for ocean modelling*, Ocean Dynamics, 57 (2007), pp. 109–121.

[14] S. Blaise and A. St-Cyr, *A dynamic hp-adaptive discontinuous Galerkin method for shallow-water flows on the sphere with application to a global tsunami simulation*, Monthly Weather Rev., 140 (2012), pp. 978–996.

[15] P. Bochev and M. Shashkov, *Constrained interpolation (remap) of divergence-free fields*, Comput. Methods Appl. Mech. Engrg., 194 (2005), pp. 511–530.

[16] H. Borouchaki, F. Hecht, and P. J. Frey, *Mesh gradation control*, Int. J. Numer. Methods Engrg., 43 (1998), pp. 1143–1165, https://doi.org/10.1002/(SICI)1097-0207(19981130)43:6⟨1143::AID-NME470⟩3.0.CO;2-I,

[17] F. Botelho, *Functional Analysis and Applied Optimization in Banach Spaces: Applications to Non-convex Variational Models*, Springer, New York, 2014.

[18] F. Brezzi, *On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers*, Rev. Française Automat. Informat. Recherche Opérationelle Sér. Rouge, 8 (1974), pp. 129–151.

[19] A. G. Buchan, P. E. Farrell, G. J. Gorman, A. J. H. Goddard, M. D. Eaton, E. T. Nygaard, P. L. Angelo, R. P. Smedley-Stevenson, S. R. Merton, and P. N. Smith, *The immersed body supermeshing method for modelling reactor-physics problems with complex internal structures*, Ann. Nuclear Energy, 63 (2014), pp. 399–408.

[20] G. F. Carey, G. Bicken, V. Carey, C. Berger, and J. Sanchez, *Locally constrained projections on grids*, Int. J. Numer. Methods Engrg., 50 (2001), pp. 549–577.

[21] L. Chen, P. Sun, and J. Xu, *Optimal anisotropic meshes for minimizing interpolation errors in L^p-norm*, Math. Comp., 76 (2007), pp. 179–204, https://doi.org/10.1090/S0025-5718-06-01896-5.

[22] T. J. Chung, *Finite Element Analysis in Fluid Dynamics*, McGraw-Hill, New York, 1978.

[23] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*, Classics in Appl. Math. 40, SIAM, Philadelphia, 2002, https://doi.org/10.1137/1.9780898719208.

[24] C. J. Cotter and D. A. Ham, *Numerical wave propagation for the triangular $P1_{DG}$–$P2$ finite element pair*, J. Comput. Phys., 230 (2011), pp. 2806–2820.

[25] C. J. Cotter, D. A. Ham, and C. C. Pain, *A mixed discontinuous/continuous finite element pair for shallow-water ocean modelling*, Ocean Model., 26 (2009), pp. 86–90.

[26] C. J. Cotter, D. A. Ham, C. C. Pain, and S. Reich, *LBB stability of a mixed Galerkin finite element pair for fluid flow simulations*, J. Comput. Phys., 228 (2009), pp. 336–348.

[27] C. J. Cotter and J. Shipton, *Mixed finite elements for numerical weather prediction*, J. Comput. Phys., 231 (2012), pp. 7076–7091.

[28] C. J. Cotter and J. Thuburn, *A finite element exterior calculus framework for the rotating shallow-water equations*, J. Comput. Phys., 257 (2014), pp. 1506–1526.

[29] S. Danilov, *Ocean modeling on unstructured meshes*, Ocean Model., 69 (2013), pp. 195–210.

[30] D. R. Davies, C. R. Wilson, and S. C. Kramer, *Fluidity: A fully unstructured anisotropic adaptive mesh computational modeling framework for geodynamics*, Geochem. Geophys. Geosyst., 12 (2011), Q06001.

[31] D. H. Eberly, *3D Game Engine Design: A Practical Approach to Real-Time Computer Graphics*, 2nd ed., CRC Press, Boca Raton, FL, 2007.

[32] C. Eskilsson, *An hp-adaptive discontinuous Galerkin method for shallow water flows*, Int. J. Numer. Methods Fluids, 67 (2011), pp. 1605–1623.

[33] R. D. Falgout and U. M. Yang, *hypre: A library of high performance preconditioners*, in Proceedings of Computational Science – ICCS 2002, P. M. A. Sloot et al., eds., Lecture Notes in Comput. Sci. 2331, Springer, Berlin, 2002, pp. 632–641.

[34] P. E. Farrell, *Galerkin Projection of Discrete Fields via Supermesh Construction*, Ph.D. thesis, Imperial College London, 2009.

[35] P. E. Farrell, *The addition of fields on different meshes*, J. Comput. Phys., 230 (2011), pp. 3265–3269.

[36] P. E. Farrell and J. R. Maddison, *Conservative interpolation between volume meshes by local Galerkin projection*, Comput. Methods Appl. Mech. Engrg., 200 (2011), pp. 89–100.
[37] P. E. Farrell, M. D. Piggott, C. C. Pain, G. J. Gorman, and C. R. Wilson, *Conservative interpolation between unstructured meshes via supermesh construction*, Comput. Methods Appl. Mech. Engrg., 198 (2009), pp. 2632–2642.
[38] G. J. Fix, *Finite element models for ocean circulation problems*, SIAM J. Appl. Math., 29 (1975), pp. 371–387, https://doi.org/10.1137/0129031.
[39] R. Ford, C. C. Pain, M. D. Piggott, A. J. H. Goddard, C. R. E. de Oliveira, and A. P. Umpleby, *A nonhydrostatic finite-element model for three-dimensional stratified oceanic flows. Part* I: *Model formulation*, Monthly Weather Rev., 132 (2004), pp. 2816–2831.
[40] M. J. Gander and C. Japhet, *An algorithm for non-matching grid projections with linear complexity*, in Domain Decomposition Methods in Science and Engineering XVIII, M. Bercovier et al., eds., Lecture Notes in Comput. Sci. Eng. 70, Springer, Berlin, 2009, pp. 185–192.
[41] M. J. Gander and C. Japhet, *Algorithm* 932*: PANG: Software for nonmatching grid projections in* 2*D and* 3*D with linear complexity*, ACM Trans. Math. Software, 40 (2013), 6.
[42] P. L. George and H. Borouchaki, *Delaunay Triangulation and Meshing: Application to Finite Elements*, Hermes, Paris, 1998.
[43] C. Geuzaine and J.-F. Remacle, *Gmsh: A* 3*-D finite element mesh generator with built-in pre- and post-processing facilities*, Int. J. Numer. Methods Engrg., 79 (2009), pp. 1309–1331.
[44] G. J. Gorman, C. C. Pain, M. D. Piggott, A. P. Umpleby, P. E. Farrell, and J. R. Maddison, *Interleaved parallel tetrahedral mesh optimisation and dynamic load-balancing*, in Proceedings of the International Conference on Adaptive Modeling and Simulation (ADMOS 2009), Brussells, Belgium, P. Bouillard and P. Díez, eds., 2009, pp. 101–104.
[45] J. Grandy, *Conservative remapping and region overlays by intersecting arbitrary polyhedra*, J. Comput. Phys., 148 (1999), pp. 433–466.
[46] P. M. Gresho and R. L. Sani, *Incompressible Flow and the Finite Element Method. Vol.* 1*: Advection-Diffusion*, John Wiley & Sons, New York, 2000.
[47] P. M. Gresho and R. L. Sani, *Incompressible Flow and the Finite Element Method. Vol.* 2*: Isothermal Laminar Flow,* John Wiley & Sons, New York, 2000.
[48] S. M. Griffies, A. J. Adcroft, H. Banks, C. W. Böning, E. P. Chassignet, G. Danabasoglu, S. Danilov, E. Deleersnijder, H. Drange, M. England, B. Fox-Kemper, R. Gerdes, A. Gnanadesikan, R. J. Greatbatch, R. W. Hallberg, E. Hanert, M. J. Harrison, S. Legg, C. M. Little, G. Madec, S. J. Marsland, M. Nikurashin, A. Pirani, H. L. Simmons, J. Schröter, B. L. Samuels, A.-M. Treguier, J. R. Toggweiler, H. Tsujino, G. K. Vallis, and L. White, *Problems and prospects in large-scale ocean circulation models*, in Proceedings of OceanObs'09: Sustained Ocean Observations and Information for Society, (Venice, Italy), vol. 2, J. Hall, D. E. Harrison, and D. Stammer, eds., ESA Publication WPP-306, 2010.
[49] S. M. Griffies, C. Böning, F. O. Bryan, E. P. Chassignet, R. Gerdes, H. Hasumi, A. Hirst, A. M. Treguier, and D. Webb, *Developments in ocean climate modelling*, Ocean Model., 2 (2000), pp. 123–192.
[50] J. L. Guermond, P. Minev, and J. Shen, *An overview of projection methods for incompressible flows*, Comput. Methods Appl. Mech. Engrg., 195 (2006), pp. 6011–6045.
[51] C. Härtel, E. Meiburg, and F. Necker, *Analysis and direct numerical simulation of the flow at a gravity current head. Part* 1*. Flow topology and front speed for slip and no-slip boundaries*, J. Fluid Mech., 418 (2000), pp. 189–212, https://doi.org/10.1017/S0022112000001221.
[52] V. E. Henson and U. M. Yang, *BoomerAMG: A parallel algebraic multigrid solver and preconditioner*, Appl. Numer. Math., 41 (2002), pp. 155–177.
[53] H. R. Hiester, *The Application of Adaptive Mesh Techniques to Numerical Simulations of Gravity Current Flows*, PhD thesis, Imperial College London, 2011.
[54] H. R. Hiester, M. D. Piggott, and P. A. Allison, *The impact of mesh adaptivity on the gravity current front speed in a two-dimensional lock-exchange*, Ocean Model., 38 (2011), pp. 1–21.
[55] H. R. Hiester, M. D. Piggott, P. E. Farrell, and P. A. Allison, *Assessment of spurious mixing in adaptive mesh simulations of the two-dimensional lock-exchange*, Ocean Model., 73 (2014), pp. 30–44.
[56] R. C. Kirby, *Algorithm* 839*: FIAT, a new paradigm for computing finite element basis functions*, ACM Trans. Math. Software, 30 (2004), pp. 502–516.

[57] R. C. Kirby and A. Logg, *A compiler for variational forms*, ACM Trans. Math. Software, 32 (2006), pp. 417–444.
[58] S. C. Kramer, C. J. Cotter, and C. C. Pain, *Solving the Poisson equation on small aspect ratio domains using unstructured meshes*, Ocean Model., 35 (2010), pp. 253–263.
[59] O. A. Ladyzhenskaya, *The Mathematical Theory of Viscous Incompressible Flow*, second English edition, revised and enlarged, Math. Appl. 2, translated from the Russian by R. A. Silverman and J. Chu, Gordon and Breach Science Publishers, New York, 1969.
[60] R. J. LeVeque, *Finite–Volume Methods for Hyperbolic Problems*, Cambridge University Press, Cambridge, UK, 2002.
[61] A. Logg, K.-A. Mardal, and G. N. Wells, eds., *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book*, Lecture Notes in Comput. Sci. Engrg. 84, Springer, Berlin, 2012.
[62] A. Logg and G. N. Wells, *DOLFIN: Automated finite element computing*, ACM Trans. Math. Software, 37 (2010), 20.
[63] A. Loseille and F. Alauzet, *Continuous mesh framework Part* II*: Validations and applications*, SIAM J. Numer. Anal., 49 (2011), pp. 61–86, https://doi.org/10.1137/10078654X.
[64] A. E. MacDonald, J. Middlecoff, T. Henderson, and J.-L. Lee, *A general method for modeling on irregular grids*, Int. J. High Performance Comput. Appl., 25 (2011), pp. 392–403.
[65] J. R. Maddison, C. J. Cotter, and P. E. Farrell, *Geostrophic balance preserving interpolation in mesh adaptive linearised shallow-water ocean modelling*, Ocean Model., 37 (2011), pp. 35–48.
[66] J. R. Maddison and P. E. Farrell, *Directional integration on unstructured meshes via supermesh construction*, J. Comput. Phys., 231 (2012), pp. 4422–4432.
[67] G. R. Markall, F. Rathgeber, L. Mitchell, N. Loriant, C. Bertolli, D. A. Ham, and P. H. J. Kelly, *Performance-portable finite element assembly using PyOP*2 *and FEniCS*, in Proceedings of Supercomputing: 28th International Supercomputing Conference (ISC 2013, Leipzig, Germany), J. M. Kunkel, T. Ludwig, and H. W. Meuer, eds., Lecture Notes in Comput. Sci. 7905, Springer, Berlin, 2013, pp. 279–289.
[68] D. R. Munday, D. P. Marshall, and M. D. Piggott, *Idealised flow past an island in a dynamically adaptive finite element model*, Ocean Dynamics, 60 (2010), pp. 835–850.
[69] K. B. Ølgaard and G. N. Wells, *Optimizations for quadrature representations of finite element tensors through automated code generation*, ACM Trans. Math. Software, 37 (2010), 8.
[70] C. C. Pain, M. D. Piggott, A. J. H. Goddard, F. Fang, G. J. Gorman, D. P. Marshall, M. D. Eaton, P. W. Power, and C. R. E. de Oliveira, *Three-dimensional unstructured mesh ocean modelling*, Ocean Model., 10 (2005), pp. 5–33.
[71] C. C. Pain, A. P. Umpleby, C. R. E. De Oliveira, and A. J. H. Goddard, *Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations*, Comput. Methods Appl. Mech. Engrg., 190 (2001), pp. 3771–3796.
[72] M. D. Piggott, P. E. Farrell, C. R. Wilson, G. J. Gorman, and C. C. Pain, *Anisotropic mesh adaptivity for multi-scale ocean modelling*, Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci., 367 (2009), pp. 4591–4611.
[73] M. D. Piggott, G. J. Gorman, C. C. Pain, P. A. Allison, A. S. Candy, B. T. Martin, and M. R. Wells, *A new computational framework for multi-scale ocean modelling based on adapting unstructured meshes*, Int. J. Numer. Methods Fluids, 56 (2008), pp. 1003–1015.
[74] M. D. Piggott, C. C. Pain, G. J. Gorman, D. P. Marshall, and P. D. Killworth, *Unstructured adaptive meshes for ocean modeling*, in Ocean Modeling in an Eddying Regime, M. W. Hecht and H. Hasumi, eds., Geophys. Monograph Ser. 177, American Geophysical Union, Washington, DC, 2008, pp. 383–408.
[75] M. D. Piggott, C. C. Pain, G. J. Gorman, P. W. Power, and A. J. H. Goddard, *h, r, and hr adaptivity with applications in numerical ocean modelling*, Ocean Model., 10 (2005), pp. 95–113.
[76] T. Ringler, M. Petersen, R. L. Higdon, D. Jacobsen, P. W. Jones, and M. Maltrud, *A multi-resolution approach to global ocean modeling*, Ocean Model., 69 (2013), pp. 211–232.
[77] T. D. Ringler and D. A. Randall, *A potential enstrophy and energy conserving numerical scheme for solution of the shallow-water equations on a geodesic grid*, Monthly Weather Rev., 130 (2002), pp. 1397–1410.
[78] R. Salmon, *Poisson-bracket approach to the construction of energy- and potential-enstrophy-conserving algorithms for the shallow-water equations*, J. Atmospheric Sci., 61 (2004), pp. 2016–2036.

[79] J. E. Simpson, *Gravity Currents in the Environment and the Laboratory*, Ellis Horwood Limited, Hempstead, UK, 1987.
[80] J. Southern, G. J. Gorman, M. D. Piggott, and P. E. Farrell, *Parallel anisotropic mesh adaptivity with dynamic load balancing for cardiac electrophysiology*, J. Comput. Sci., 3 (2012), pp. 8–16.
[81] A. H. Stroud, *Approximate Calculation of Multiple Integrals*, Prentice-Hall, Englewood Cliffs, NJ, 1971.
[82] P. K. Sweby, *High resolution schemes using flux limiters for hyperbolic conservation laws*, SIAM J. Numer. Anal., 21 (1984), pp. 995–1011.
[83] Y.-H. Tseng and J. H. Ferziger, *Mixing and available potential energy in stratified flows*, Phys. Fluids, 13 (2001), pp. 1281–1293, https://doi.org/10.1063/1.1358307.
[84] P. Vaněk, J. Mandel, and M. Brezina, *Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems*, Computing, 56 (1996), pp. 179–196.
[85] Y. V. Vasilevskii and K. N. Lipnikov, *An adaptive algorithm for quasioptimal mesh generation*, Comput. Math. Math. Phys., 39 (1999), pp. 1468–1486.
[86] A. Viré, J. Xiang, F. Milthaler, P. E. Farrell, M. D. Piggott, J.-P. Latham, D. Pavlidis, and C. C. Pain, *Modelling of fluid-solid interactions using an adaptive mesh fluid model coupled with a combined finite-discrete element model*, Ocean Dynamics, 62 (2012), pp. 1487–1501.
[87] Q. Wang, H. Zhou, and W. Decheng, *Numerical simulation of wind turbine blade-tower interaction*, J. Marine Sci. Appl., 11 (2012), pp. 321–327.
[88] L. White, V. Legat, and E. Deleersnijder, *Tracer conservation for three-dimensional, finite-element, free-surface ocean modeling on moving prismatic meshes*, Monthly Weather Rev., 136 (2008), pp. 420–442.
[89] C. Wilson, *Modelling Multiple-Material Flows on Adaptive Unstructured Meshes*, Ph.D. thesis, Imperial College London, 2009.
[90] K. B. Winters and E. A. D'Asaro, *Diascalar flux and the rate of fluid mixing*, J. Fluid Mech., 317 (1996), pp. 179–193, https://doi.org/10.1017/S0022112096000717.
[91] K. B. Winters, P. N. Lombard, J. J. Riley, and E. A. D'Asaro, *Available potential energy and mixing in density-stratified fluids*, J. Fluid Mech., 289 (1995), pp. 115–128, https://doi.org/10.1017/S002211209500125X.